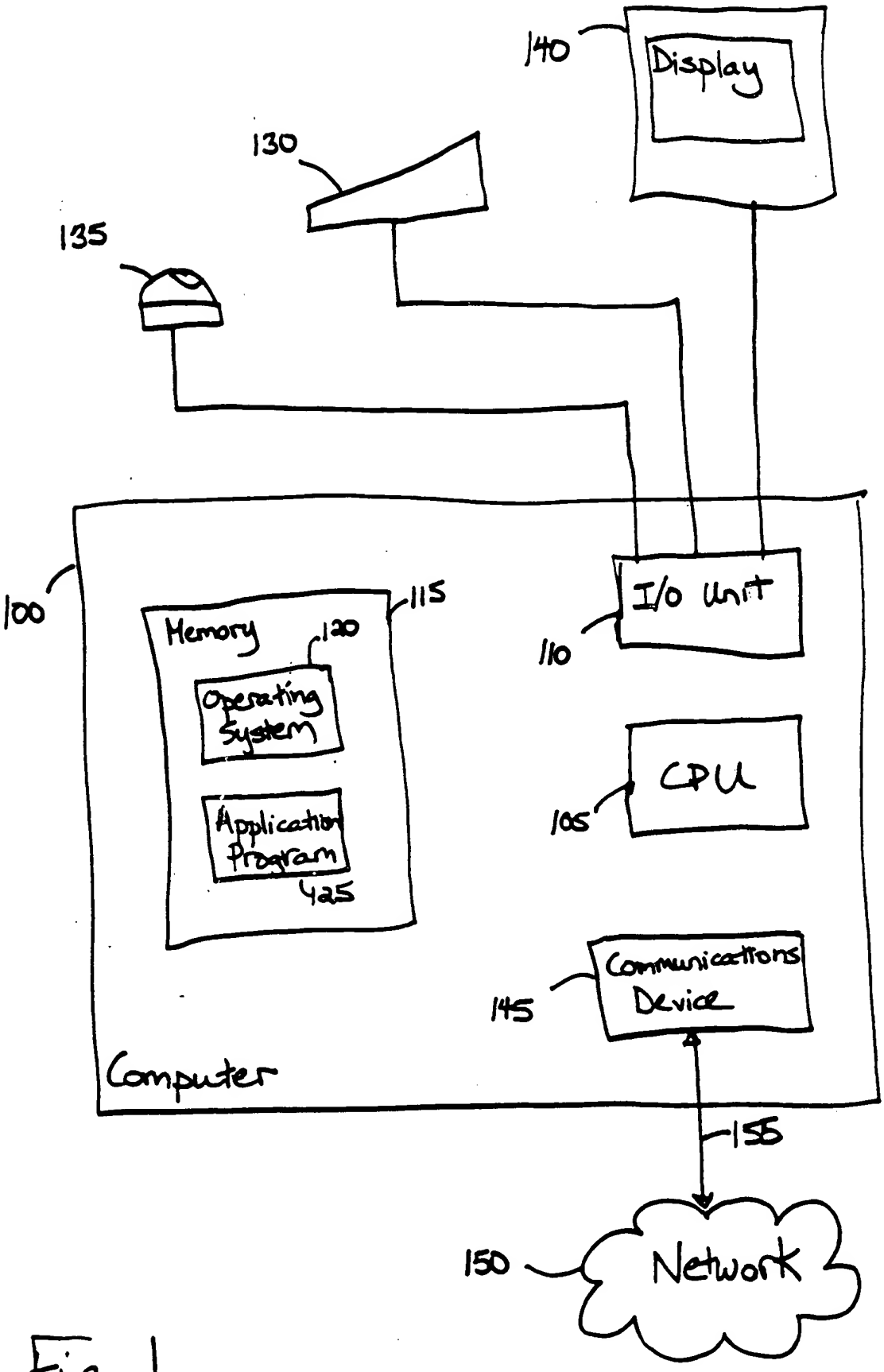

140
Display
130
135
100
115
Memory
120
Operating System
Application Program
425
I/O Unit
110
CPU
105
Communications Device
145
Computer
155
150
Network

Fig. 1

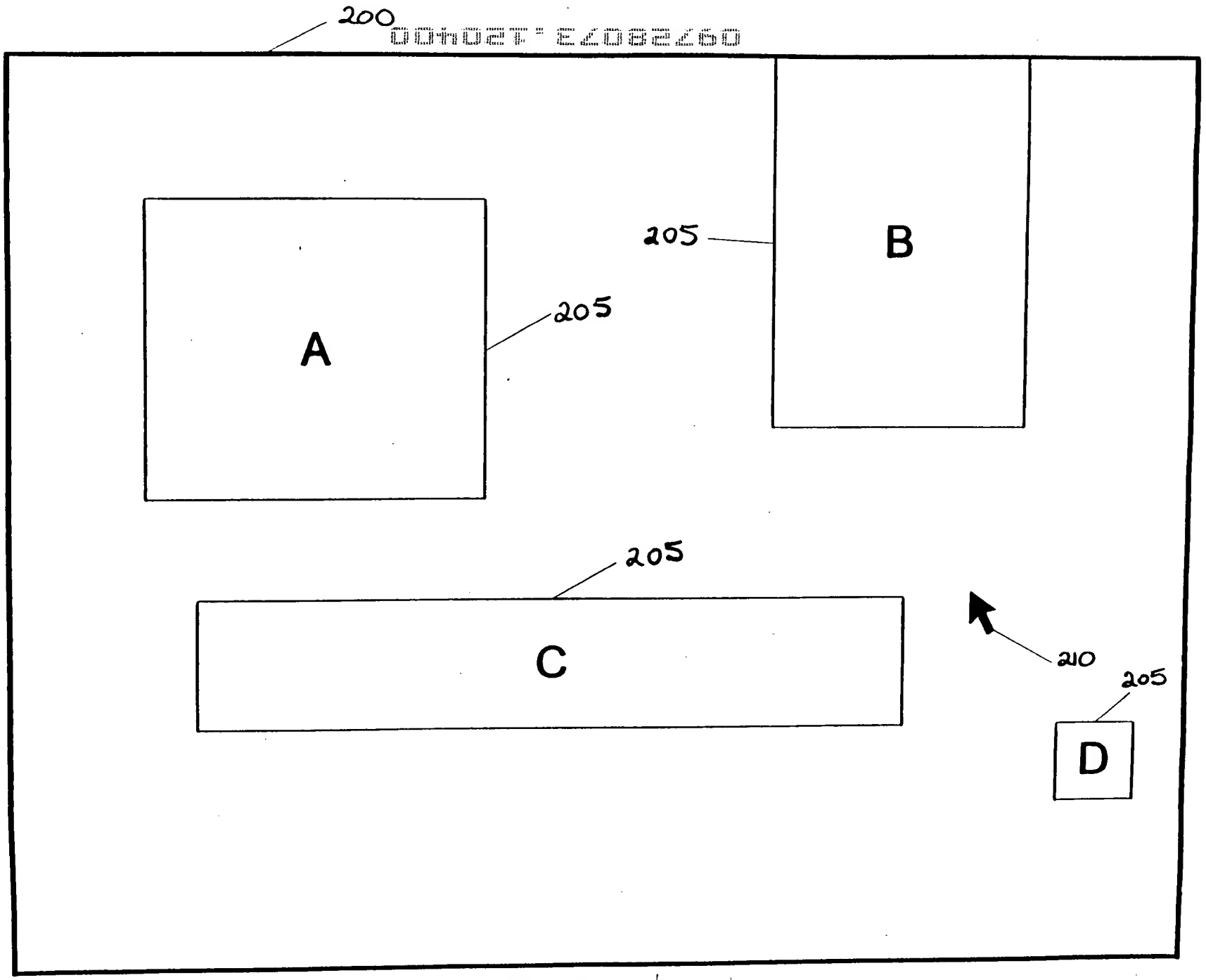


Fig. 2
PRIOR ART

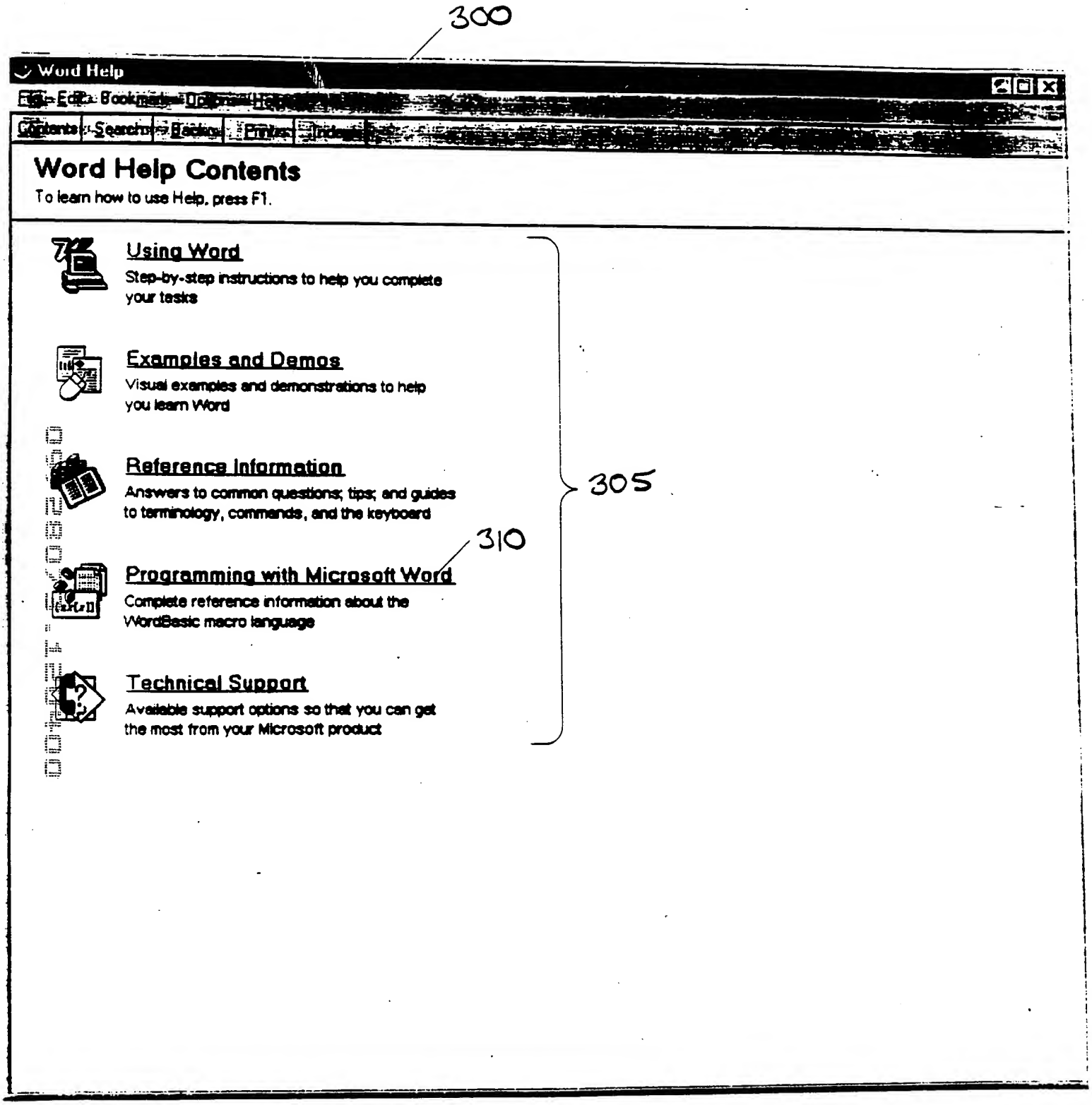


Fig. 3A
PRIOR ART

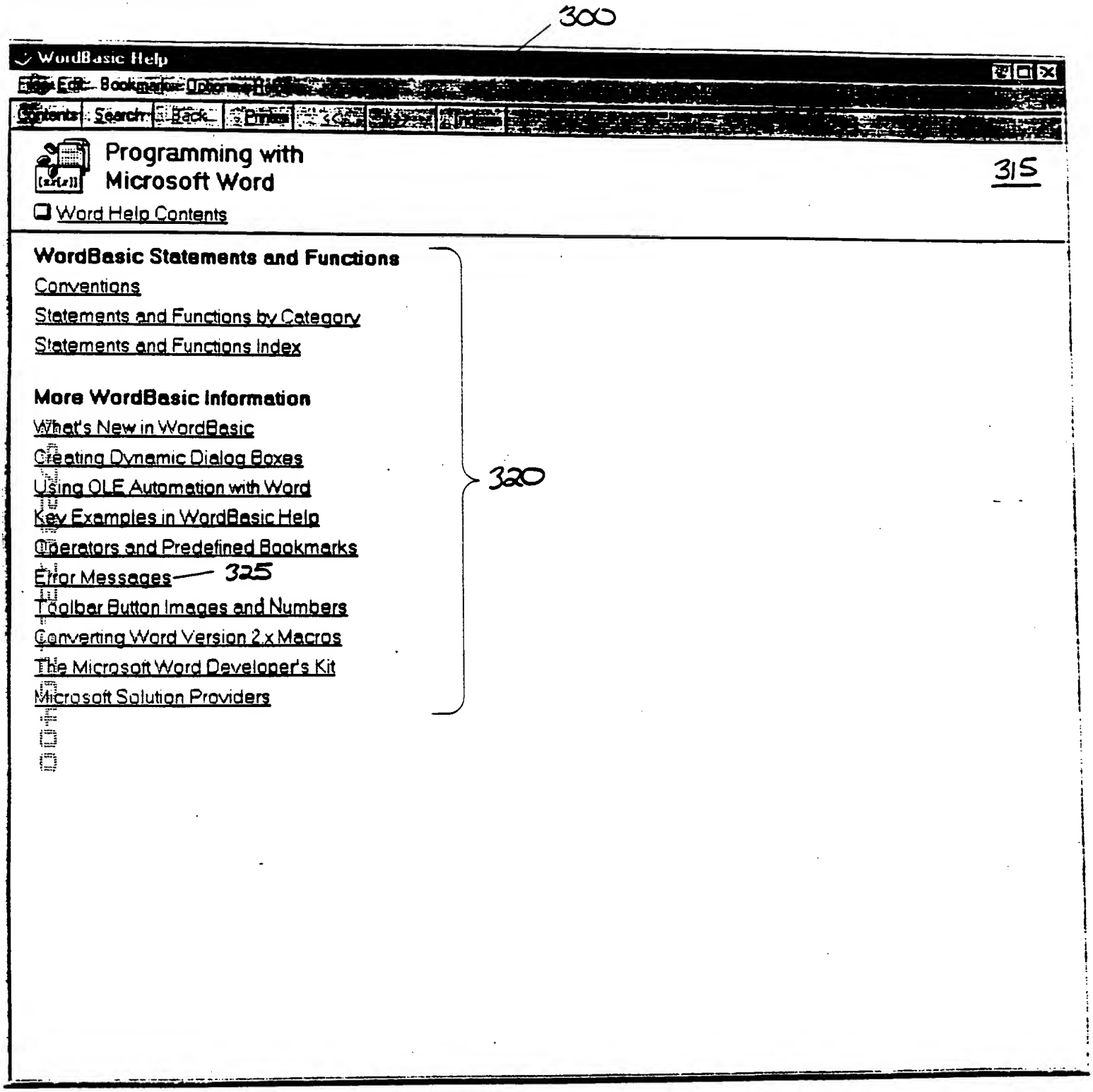


Fig. 3B
PRIOR ART

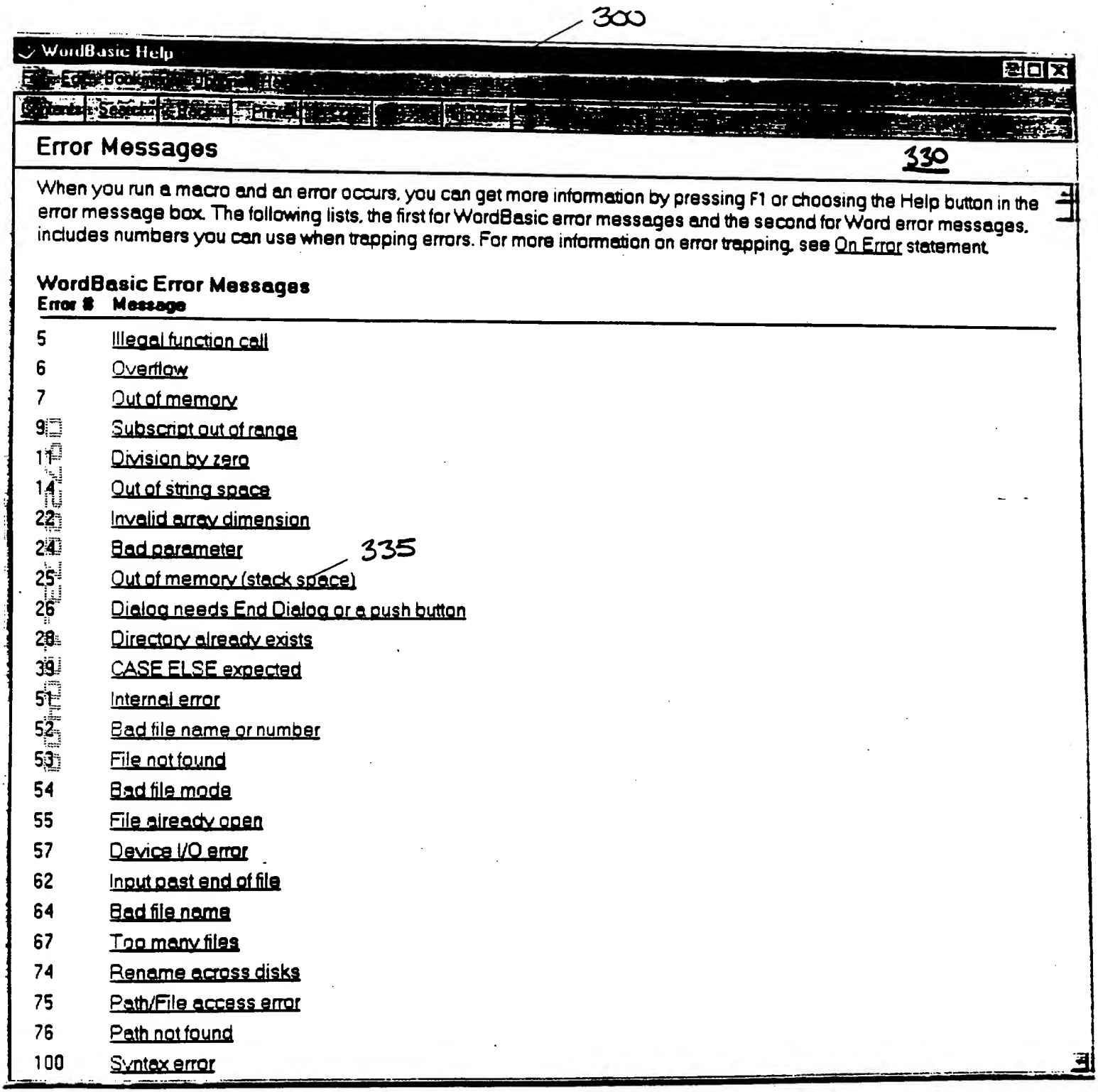


Fig. 3C
PRIOR ART

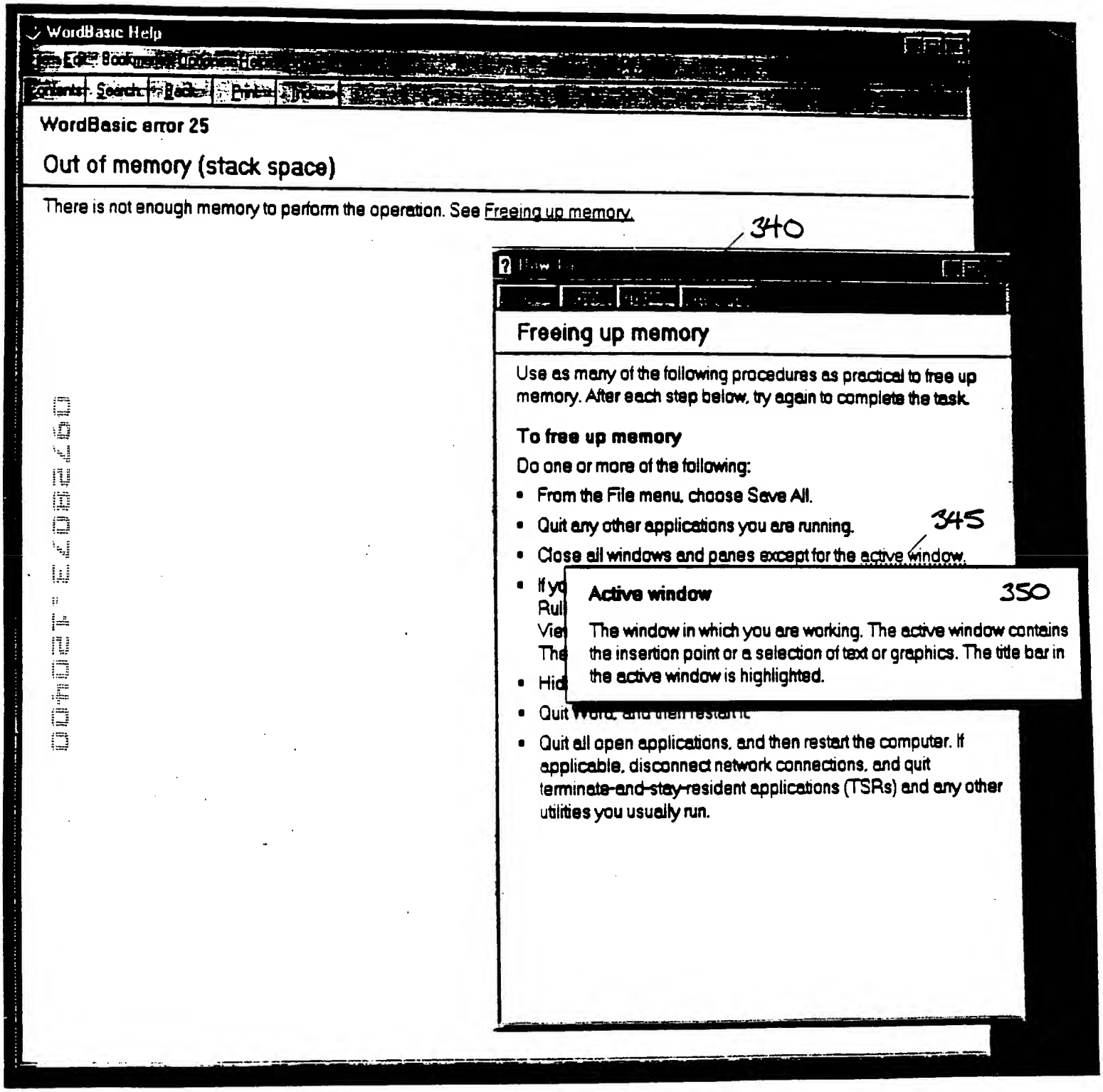


Fig. 3D
PRIOR ART

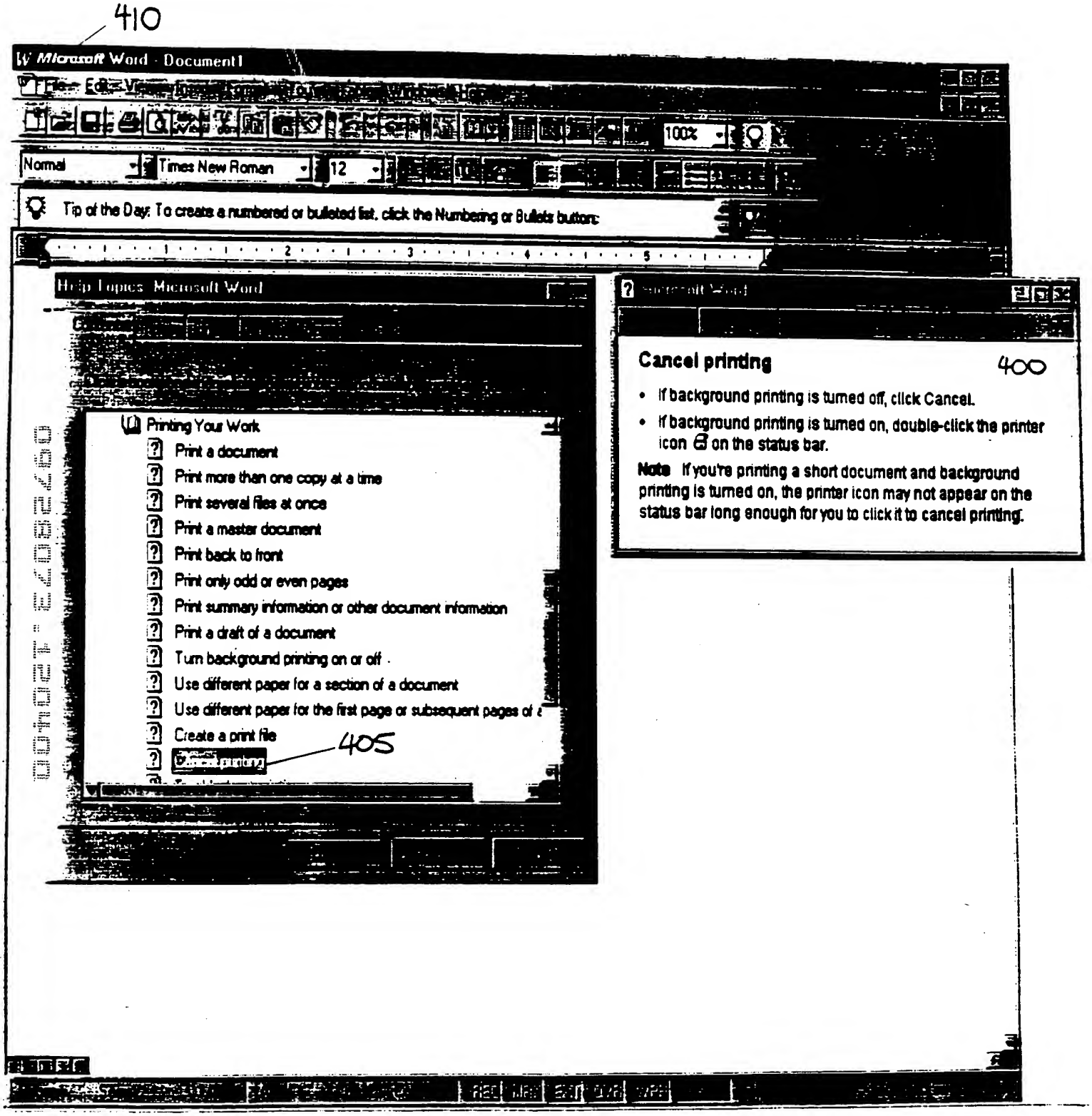


Fig. 4
PRIOR ART

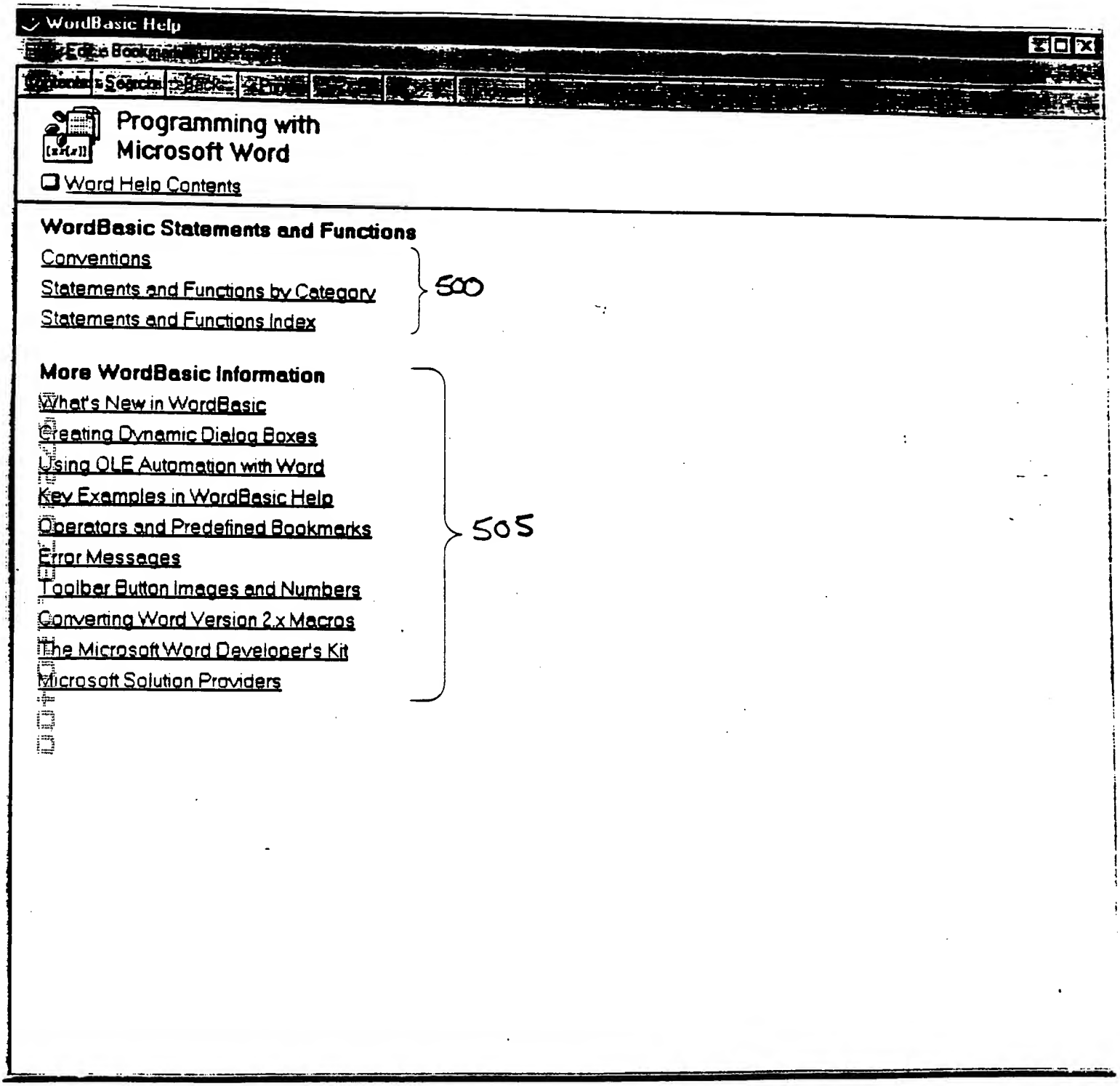


Fig. 5A
PRIOR ART

WordBasic Help

AddAddIn, AddAddIn()

Example

AddAddIn *AddIn$* [, *Load*]

AddAddIn(*AddIn$* [, *Load*])

The **AddAddIn** statement adds a template or Word add-in library (WLL) to the list of global templates and add-ins in the Templates And Add-ins dialog box (Templates command, File menu).

Argument	Explanation
AddIn$	The path and filename of the template or WLL
Load	Specifies whether to load the template or add-in after adding it to the list: 0 (zero) Does not load the template or add-in 1 or omitted Loads the template or add-in

The **AddAddIn()** function behaves the same as the statement and also returns a value corresponding to the position of the global template or add-in in the list, where 1 is the first template or add-in, 2 is the second, and so on. This value may be used with other add-in statements and functions.

You can use functions defined in a loaded WLL in a macro. Functions that take no arguments may be used just like WordBasic statements; you can return the names of these functions using **CountMacros()** and **MacroName$()**. Functions in the WLL that take arguments must be declared using the **Declare** statement.

For more information on loading global templates and add-ins, see Chapter 31, "Customizing and Optimizing Word," in the *Microsoft Word User's Guide*. For more information on using functions in WLLs, see Chapter 9, "More WordBasic Techniques," in the Microsoft Word Developer's Kit.

See also

Documents, Templates, and AddIns Statements and Functions

AddInState()

ClearAddIns

CountAddIns()

CountMacros()

DeleteAddIn

GetAddInID()

GetAddInName$()

MacroName$()

Fig. 5B
PRIOR ART

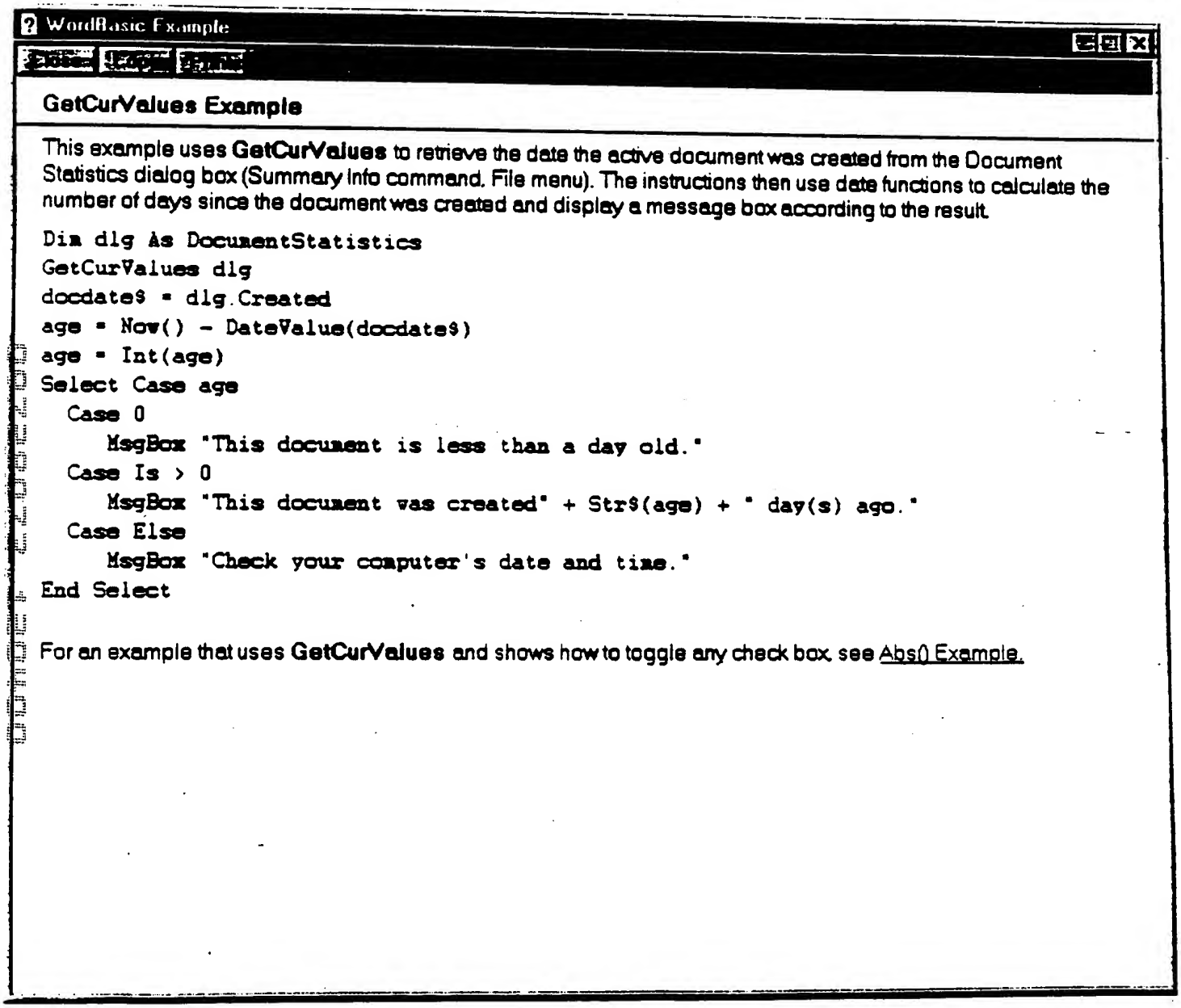


Fig. 5C
PRIOR ART

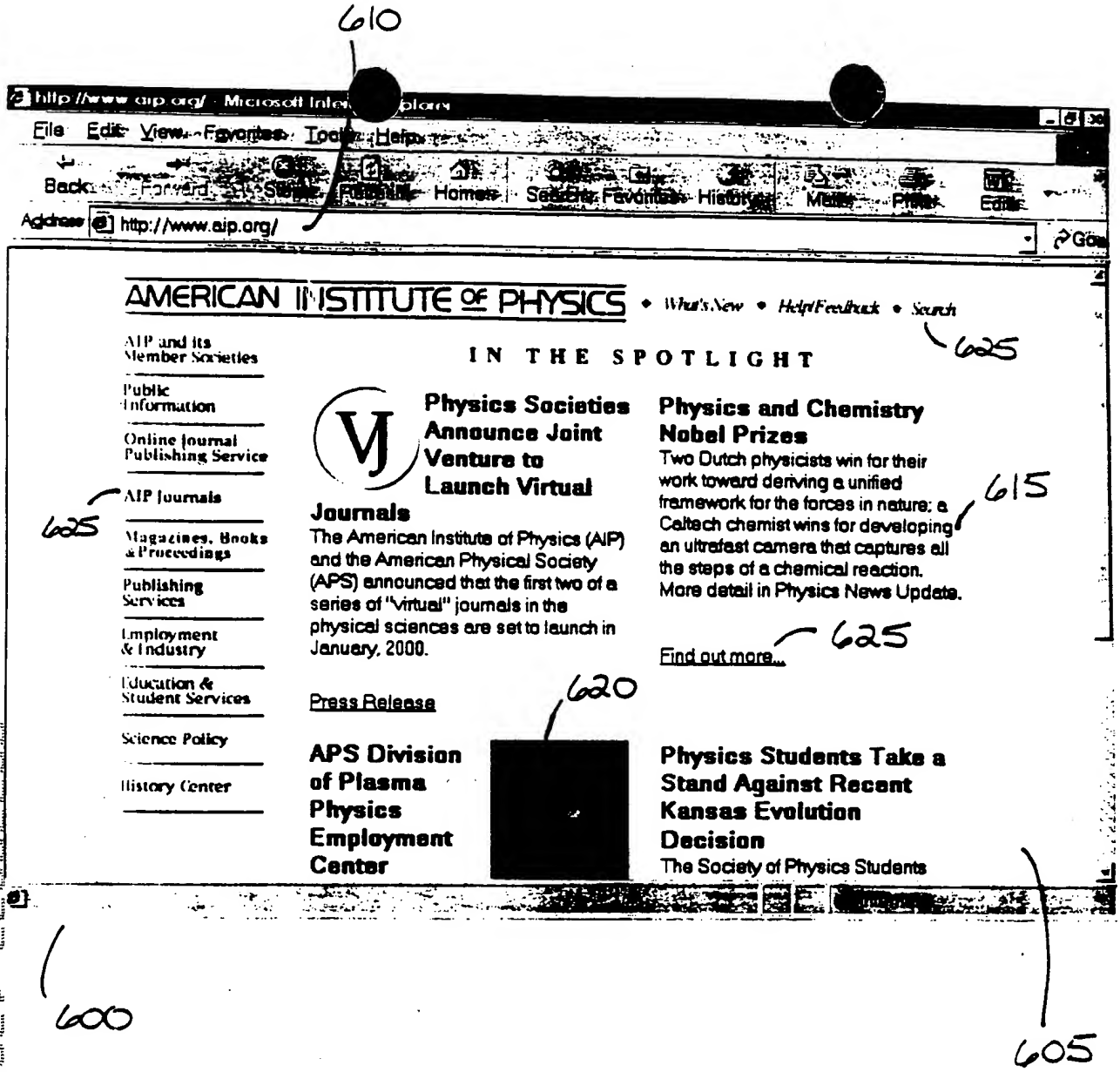

610
http://www.aip.org/
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit
Address http://www.aip.org/
AMERICAN INSTITUTE OF PHYSICS
What's New
Help/Feedback
Search
625
AIP and its Member Societies
Public Information
Online Journal Publishing Service
AIP Journals
625
Magazines, Books & Proceedings
Publishing Services
Employment & Industry
Education & Student Services
Science Policy
History Center
IN THE SPOTLIGHT
Physics Societies Announce Joint Venture to Launch Virtual Journals
The American Institute of Physics (AIP) and the American Physical Society (APS) announced that the first two of a series of "virtual" journals in the physical sciences are set to launch in January, 2000.
Press Release
620
APS Division of Plasma Physics Employment Center
Physics and Chemistry Nobel Prizes
Two Dutch physicists win for their work toward deriving a unified framework for the forces in nature; a Caltech chemist wins for developing an ultrafast camera that captures all the steps of a chemical reaction. More detail in Physics News Update.
615
Find out more...
625
Physics Students Take a Stand Against Recent Kansas Evolution Decision
The Society of Physics Students
600
605

Fig. 6

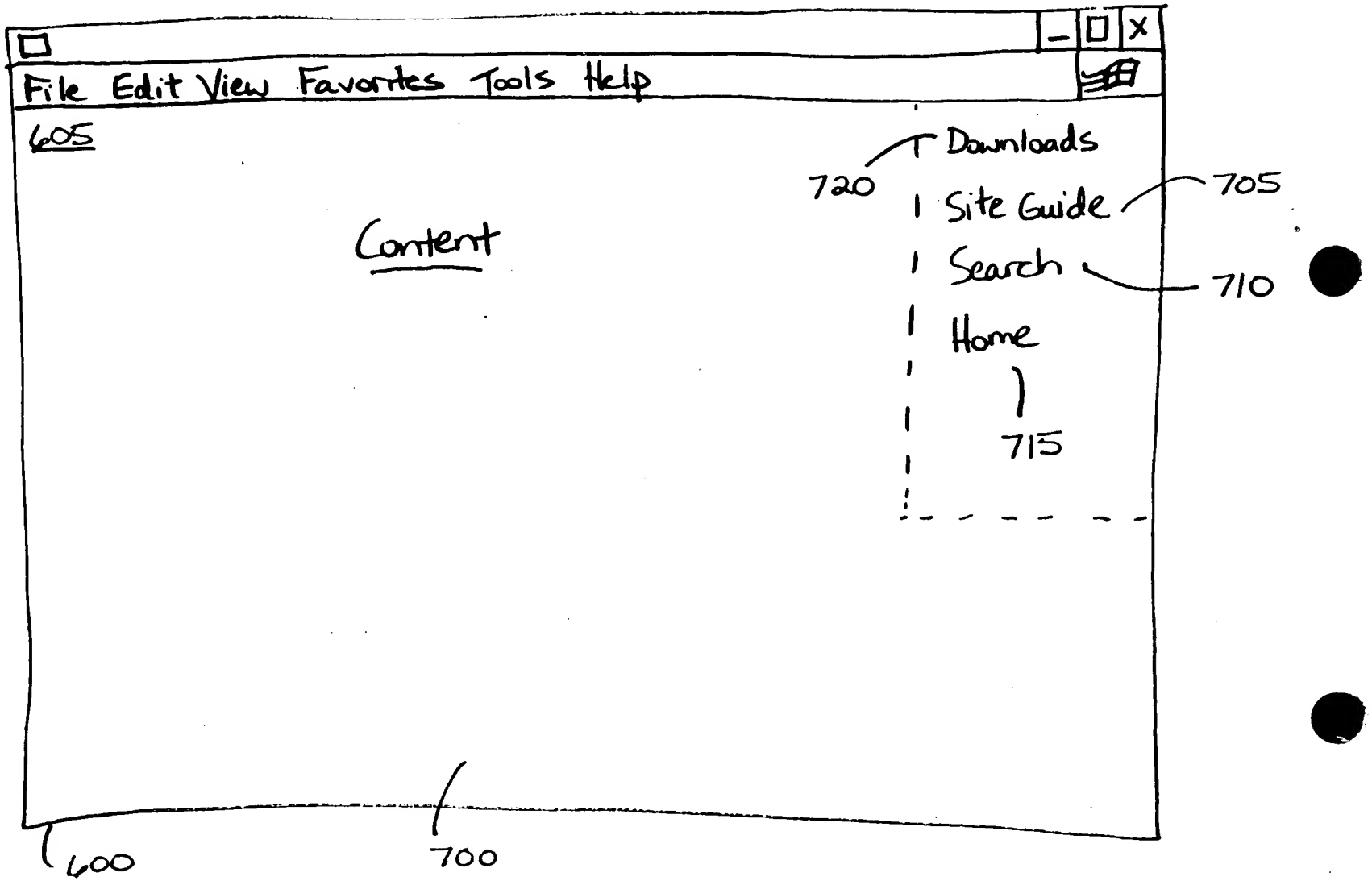


Fig. 7

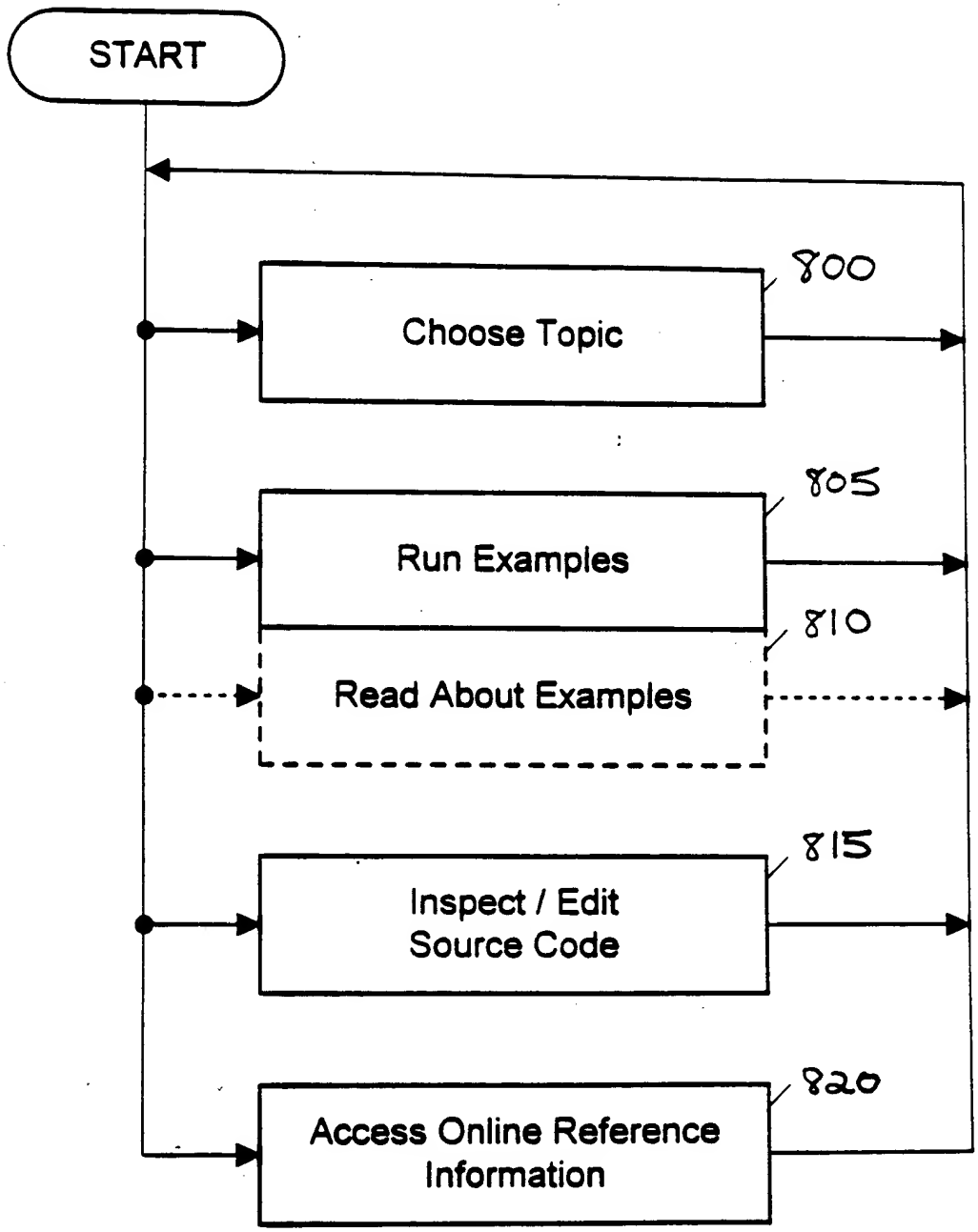

START
800
Choose Topic
805
Run Examples
810
Read About Examples
815
Inspect / Edit
Source Code
820
Access Online Reference
Information

Fig. 8

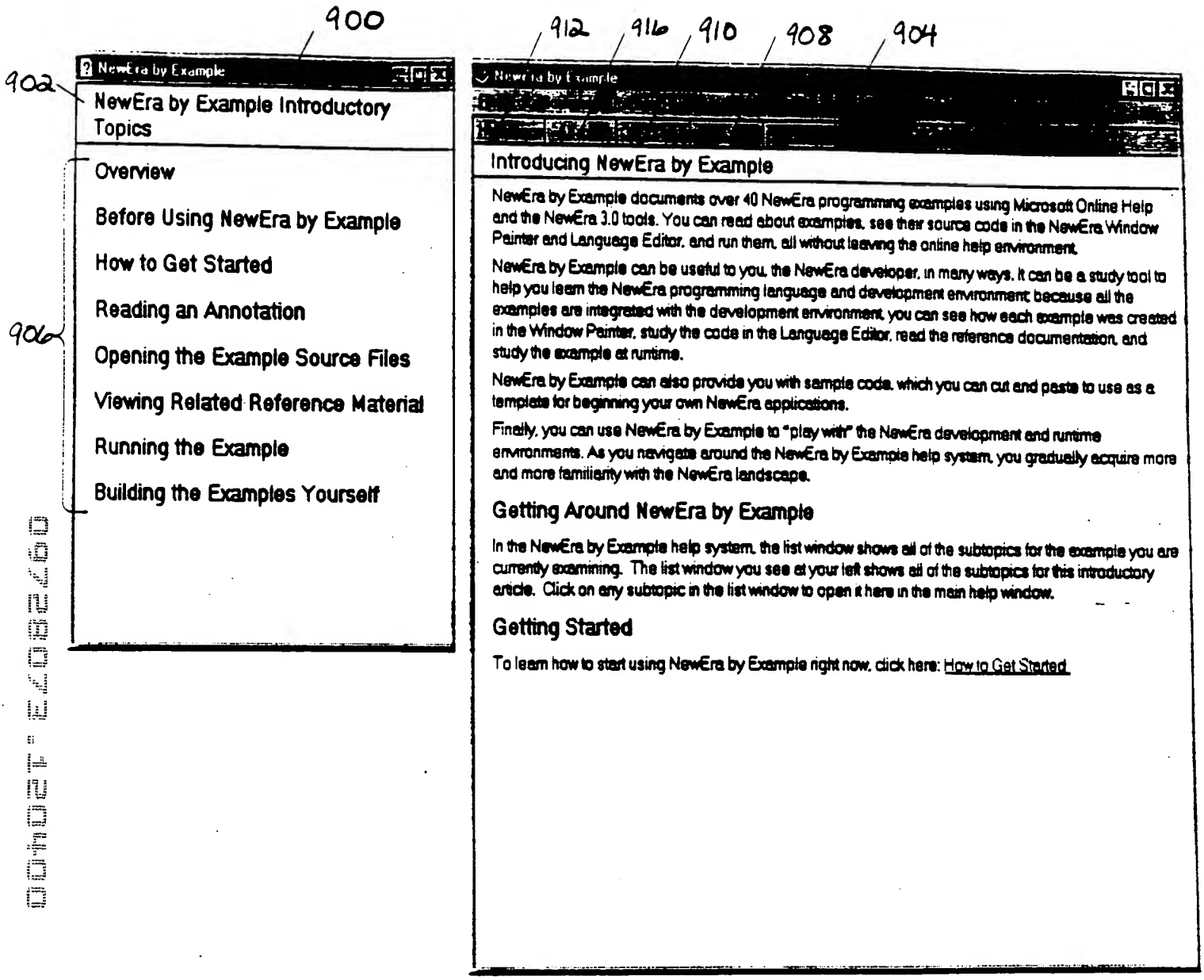

900
902
NewEra by Example
NewEra by Example Introductory Topics
Overview
Before Using NewEra by Example
How to Get Started
Reading an Annotation
Opening the Example Source Files
Viewing Related Reference Material
Running the Example
Building the Examples Yourself
906
912
916
910
908
904
Introducing NewEra by Example
NewEra by Example documents over 40 NewEra programming examples using Microsoft Online Help and the NewEra 3.0 tools. You can read about examples, see their source code in the NewEra Window Painter and Language Editor, and run them, all without leaving the online help environment.
NewEra by Example can be useful to you, the NewEra developer, in many ways. It can be a study tool to help you learn the NewEra programming language and development environment; because all the examples are integrated with the development environment, you can see how each example was created in the Window Painter, study the code in the Language Editor, read the reference documentation, and study the example at runtime.
NewEra by Example can also provide you with sample code, which you can cut and paste to use as a template for beginning your own NewEra applications.
Finally, you can use NewEra by Example to "play with" the NewEra development and runtime environments. As you navigate around the NewEra by Example help system, you gradually acquire more and more familiarity with the NewEra landscape.
Getting Around NewEra by Example
In the NewEra by Example help system, the list window shows all of the subtopics for the example you are currently examining. The list window you see at your left shows all of the subtopics for this introductory article. Click on any subtopic in the list window to open it here in the main help window.
Getting Started
To learn how to start using NewEra by Example right now, click here: How to Get Started

Fig. 9A

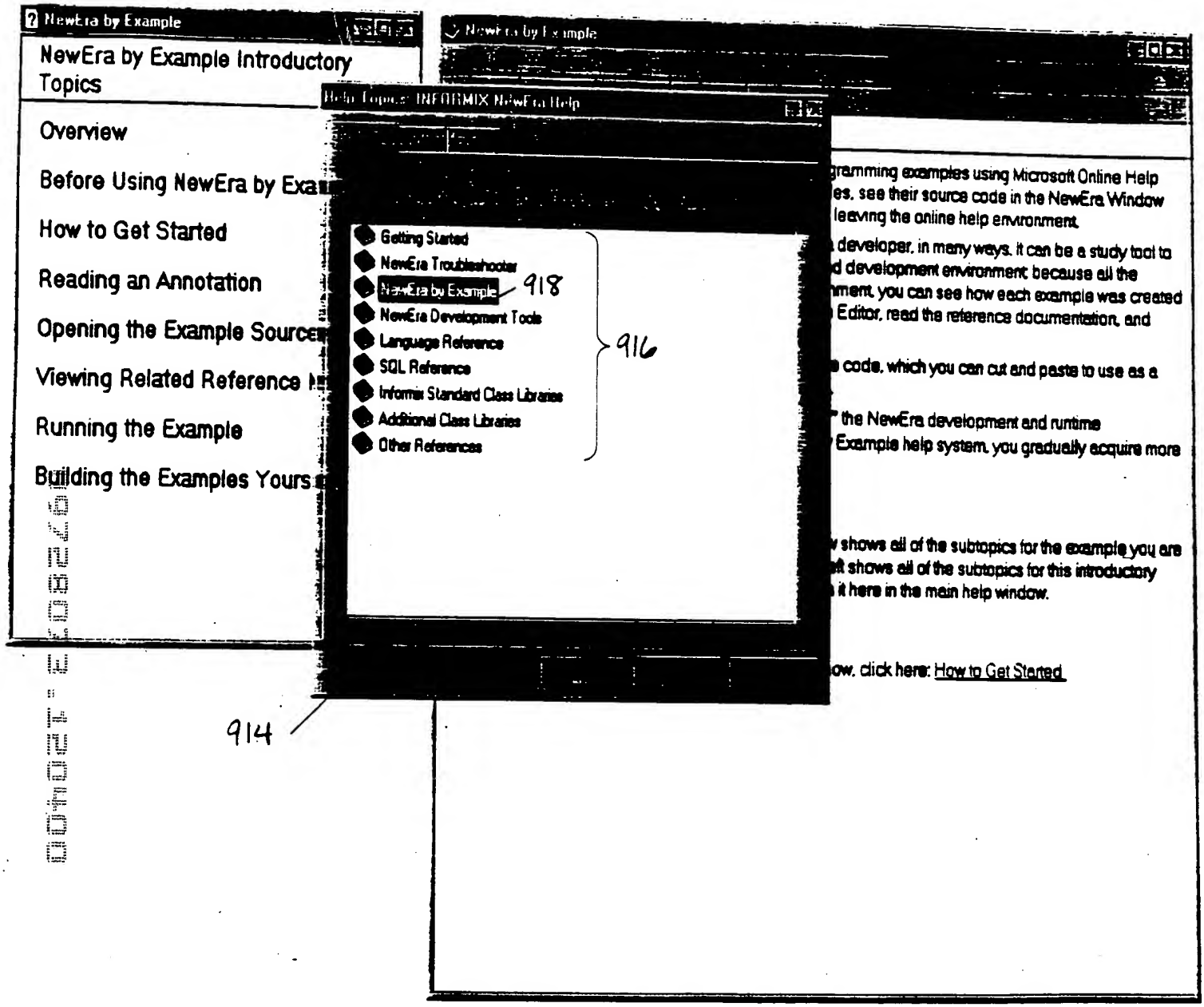

NewEra by Example
NewEra by Example Introductory Topics
Overview
Before Using NewEra by Exa
How to Get Started
Reading an Annotation
Opening the Example Source
Viewing Related Reference
Running the Example
Building the Examples Yours
Getting Started
NewEra Troubleshooter
NewEra by Example
NewEra Development Tools
Language Reference
SQL Reference
Informix Standard Class Libraries
Additional Class Libraries
Other References
918
916
914
How to Get Started

Fig. 9B

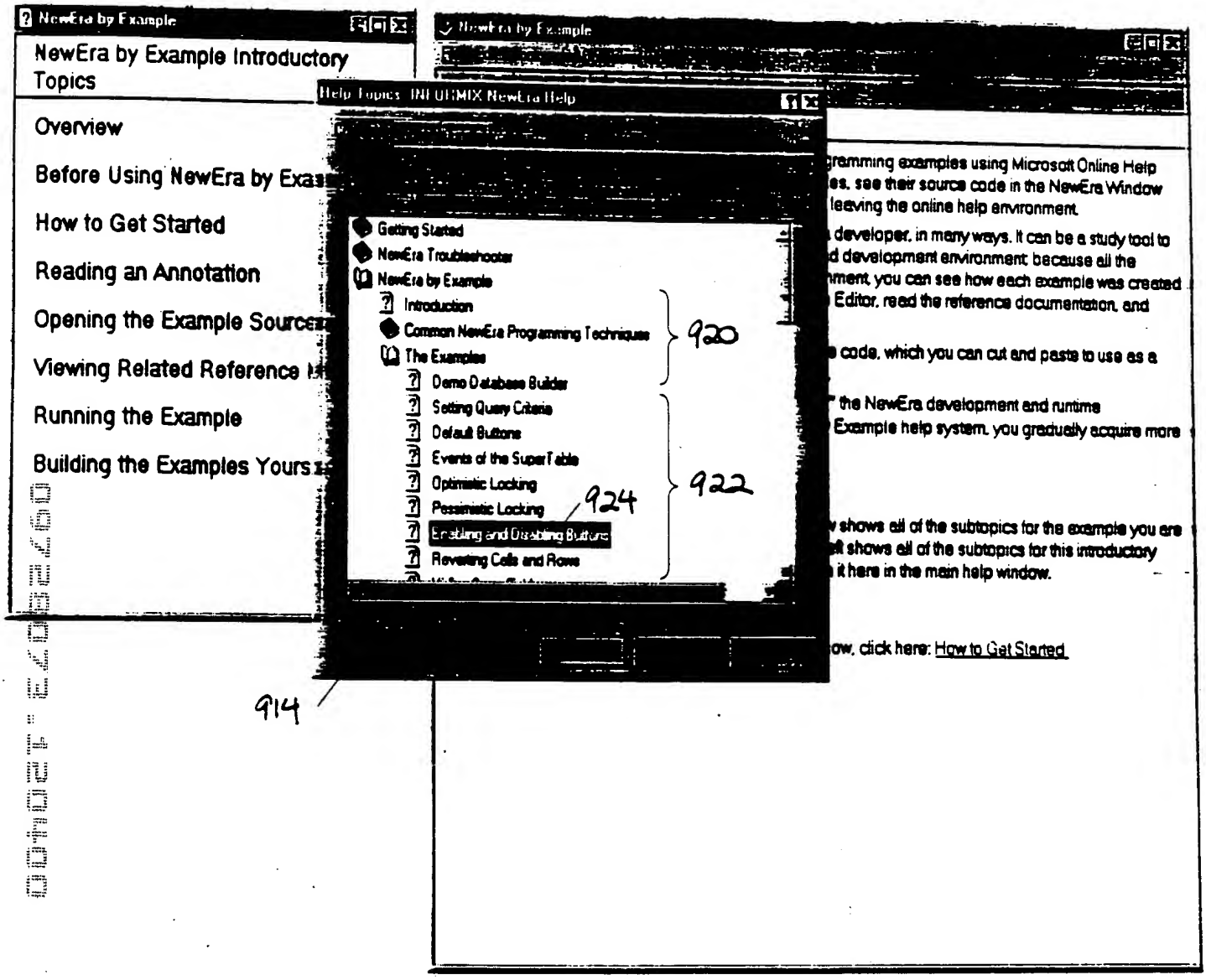

NewEra by Example
NewEra by Example Introductory Topics
Overview
Before Using NewEra by Exa
How to Get Started
Reading an Annotation
Opening the Example Source
Viewing Related Reference
Running the Example
Building the Examples Yours
Help Topics: INFORMIX NewEra Help
Getting Started
NewEra Troubleshooter
NewEra by Example
Introduction
Common NewEra Programming Techniques
The Examples
Demo Database Builder
Setting Query Criteria
Default Buttons
Events of the SuperTable
Optimistic Locking
Pessimistic Locking
Enabling and Disabling Buttons
Revealing Cells and Rows
920
922
924
914
ramming examples using Microsoft Online Help
es, see their source code in the NewEra Window
leaving the online help environment.
developer, in many ways. It can be a study tool to
d development environment because all the
nment, you can see how each example was created
Editor, read the reference documentation, and
code, which you can cut and paste to use as a
the NewEra development and runtime
Example help system, you gradually acquire more
shows all of the subtopics for the example you are
shows all of the subtopics for this introductory
it here in the main help window.
ow, click here: How to Get Started

Fig. 9C

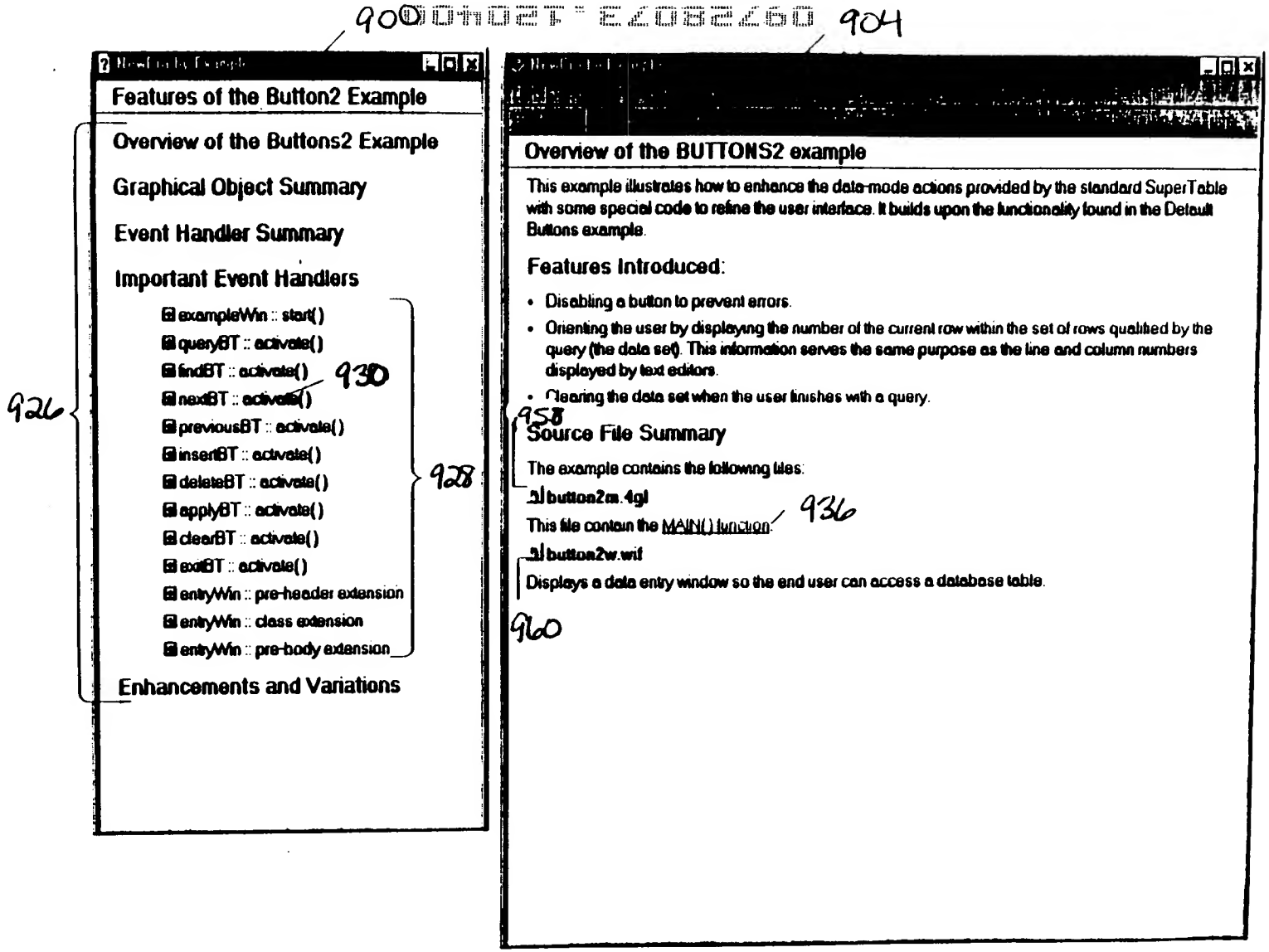

902
904
Features of the Button2 Example
Overview of the Buttons2 Example
Graphical Object Summary
Event Handler Summary
Important Event Handlers
exampleWin :: start()
queryBT :: activate()
findBT :: activate()
nextBT :: activate()
previousBT :: activate()
insertBT :: activate()
deleteBT :: activate()
applyBT :: activate()
clearBT :: activate()
exitBT :: activate()
entryWin :: pre-header extension
entryWin :: class extension
entryWin :: pre-body extension
Enhancements and Variations
926
928
930
Overview of the BUTTONS2 example
This example illustrates how to enhance the data-mode actions provided by the standard SuperTable with some special code to refine the user interface. It builds upon the functionality found in the Default Buttons example.
Features Introduced:
Disabling a button to prevent errors.
Orienting the user by displaying the number of the current row within the set of rows qualified by the query (the data set). This information serves the same purpose as the line and column numbers displayed by text editors.
Clearing the data set when the user finishes with a query.
958
Source File Summary
The example contains the following files:
button2m.4gl
This file contain the MAIN() function
936
button2w.wif
Displays a data entry window so the end user can access a database table.
960

Fig. 9D

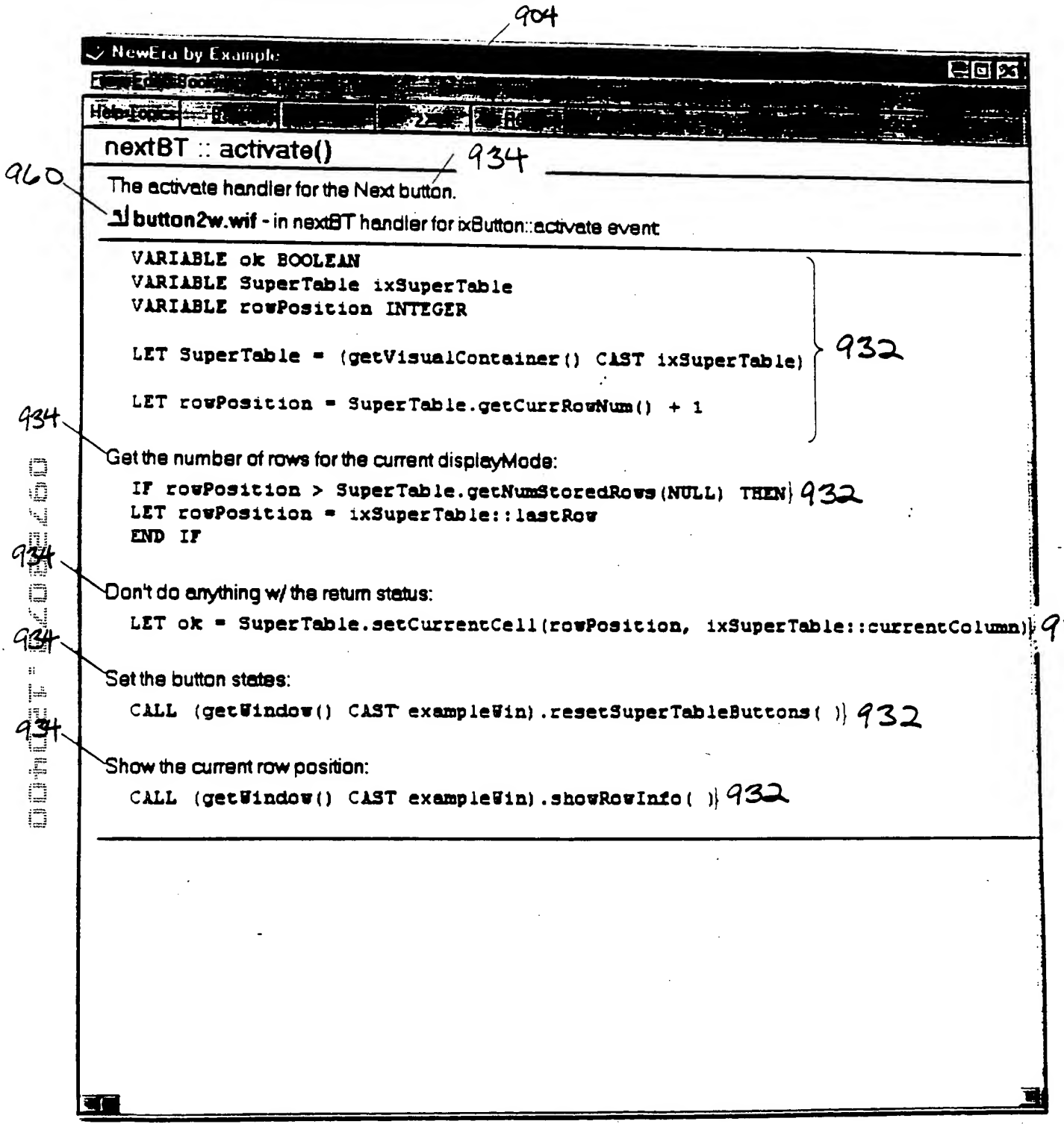

904
NewEra by Example
nextBT :: activate()
934
960
The activate handler for the Next button.
button2w.wif - in nextBT handler for ixButton::activate event
VARIABLE ok BOOLEAN
VARIABLE SuperTable ixSuperTable
VARIABLE rowPosition INTEGER
LET SuperTable = (getVisualContainer() CAST ixSuperTable)
LET rowPosition = SuperTable.getCurrRowNum() + 1
932
934
Get the number of rows for the current displayMode:
IF rowPosition > SuperTable.getNumStoredRows(NULL) THEN
LET rowPosition = ixSuperTable::lastRow
END IF
932
934
Don't do anything w/ the return status:
LET ok = SuperTable.setCurrentCell(rowPosition, ixSuperTable::currentColumn)
932
934
Set the button states:
CALL (getWindow() CAST exampleWin).resetSuperTableButtons( )
932
934
Show the current row position:
CALL (getWindow() CAST exampleWin).showRowInfo( )
932

Fig. 9E

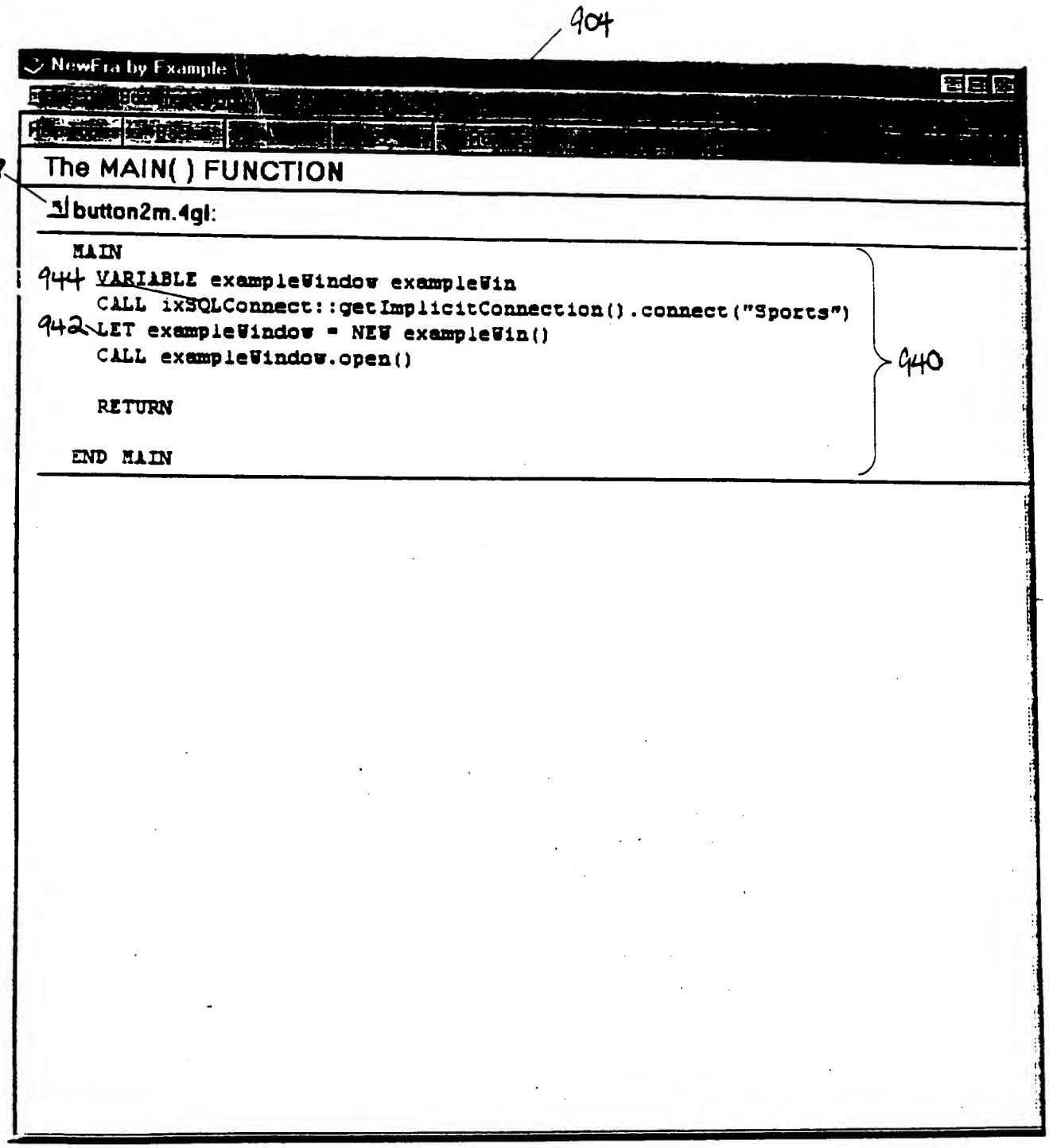

904
NewFra by Example
958
The MAIN( ) FUNCTION
button2m.4gl:
MAIN
944 VARIABLE exampleWindow exampleWin
CALL ixSQLConnect::getImplicitConnection().connect("Sports")
942 LET exampleWindow = NEW exampleWin()
CALL exampleWindow.open()
RETURN
END MAIN
940

Fig. 9F

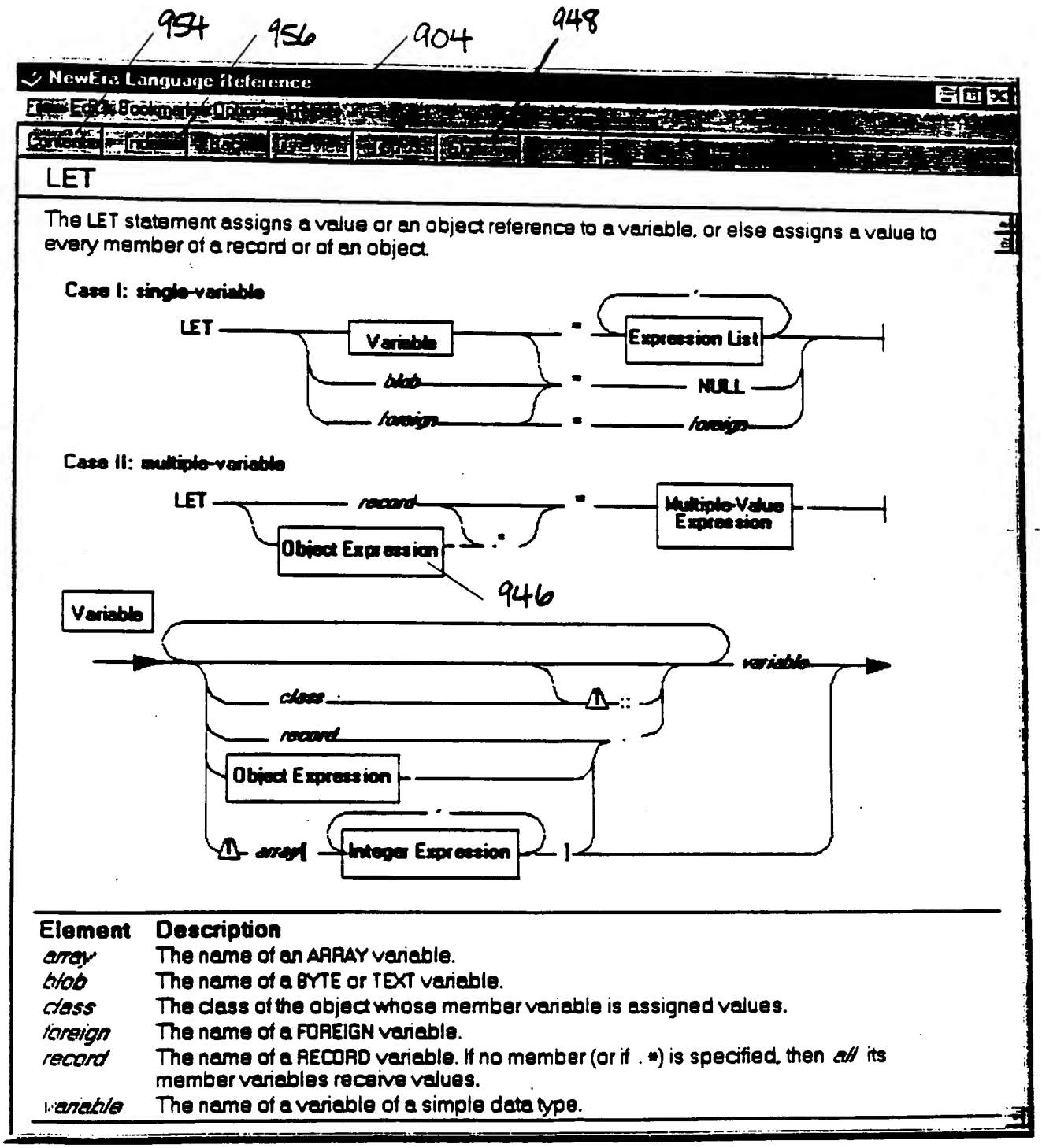

954
956
904
948
NewEra Language Reference
LET
The LET statement assigns a value or an object reference to a variable, or else assigns a value to every member of a record or of an object.
Case I: single-variable
LET
Variable
=
Expression List
blob
NULL
foreign
Case II: multiple-variable
record
Multiple-Value Expression
Object Expression
946
Variable
variable
class
::
record
Object Expression
array[
Integer Expression
]
Element
Description
array
The name of an ARRAY variable.
blob
The name of a BYTE or TEXT variable.
class
The class of the object whose member variable is assigned values.
foreign
The name of a FOREIGN variable.
record
The name of a RECORD variable. If no member (or if . *) is specified, then all its member variables receive values.
variable
The name of a variable of a simple data type.

Fig. 9G

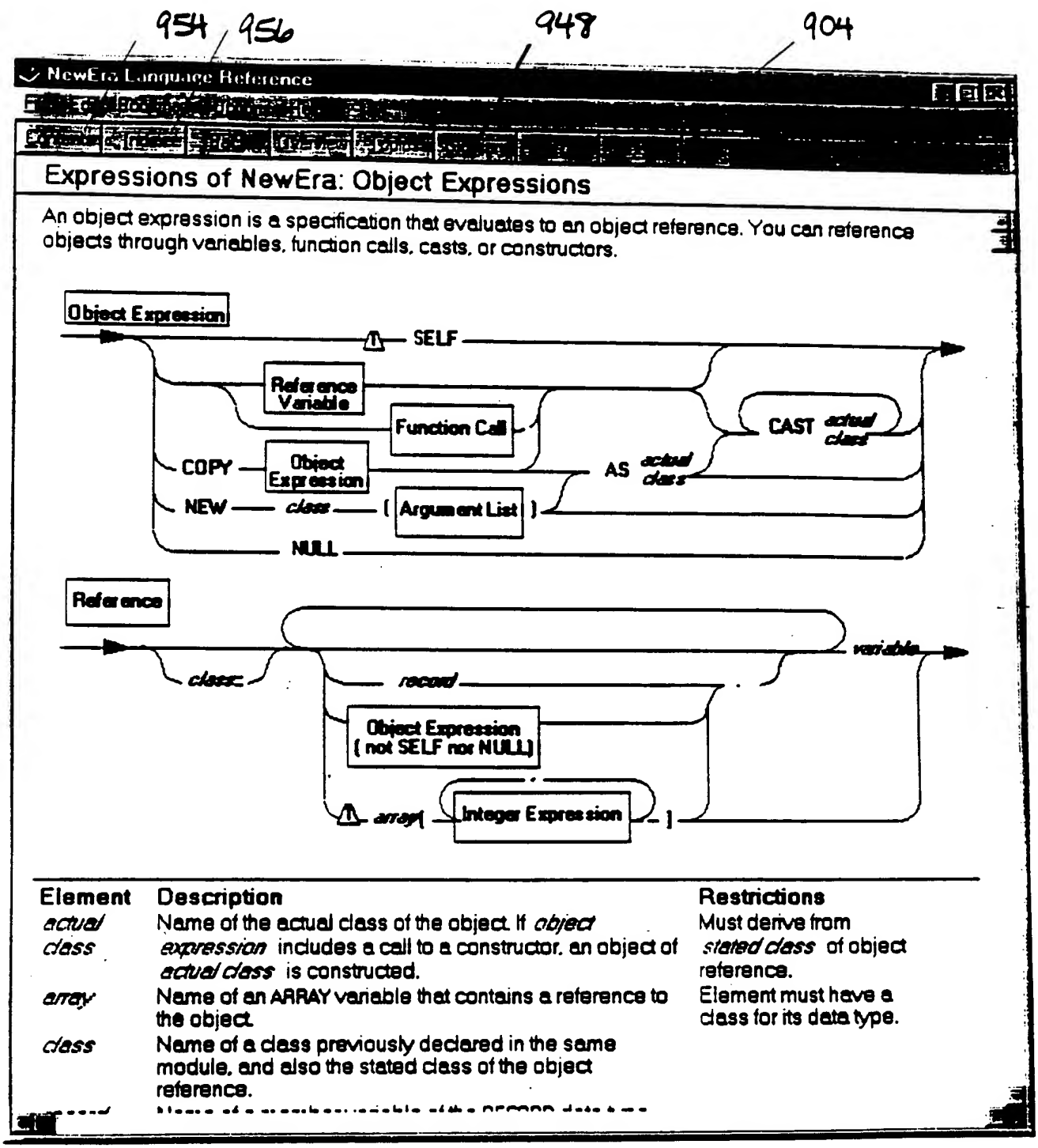

954
956
948
904
NewEra Language Reference
Expressions of NewEra: Object Expressions
An object expression is a specification that evaluates to an object reference. You can reference objects through variables, function calls, casts, or constructors.
Object Expression
SELF
Reference Variable
Function Call
CAST actual class
COPY
Object Expression
AS actual class
NEW
class
(Argument List)
NULL
Reference
class
record
variable
Object Expression (not SELF nor NULL)
array[
Integer Expression
]
Element
Description
Restrictions
actual class
Name of the actual class of the object. If object expression includes a call to a constructor, an object of actual class is constructed.
Must derive from stated class of object reference.
array
Name of an ARRAY variable that contains a reference to the object.
Element must have a class for its data type.
class
Name of a class previously declared in the same module, and also the stated class of the object reference.

Fig. 9H

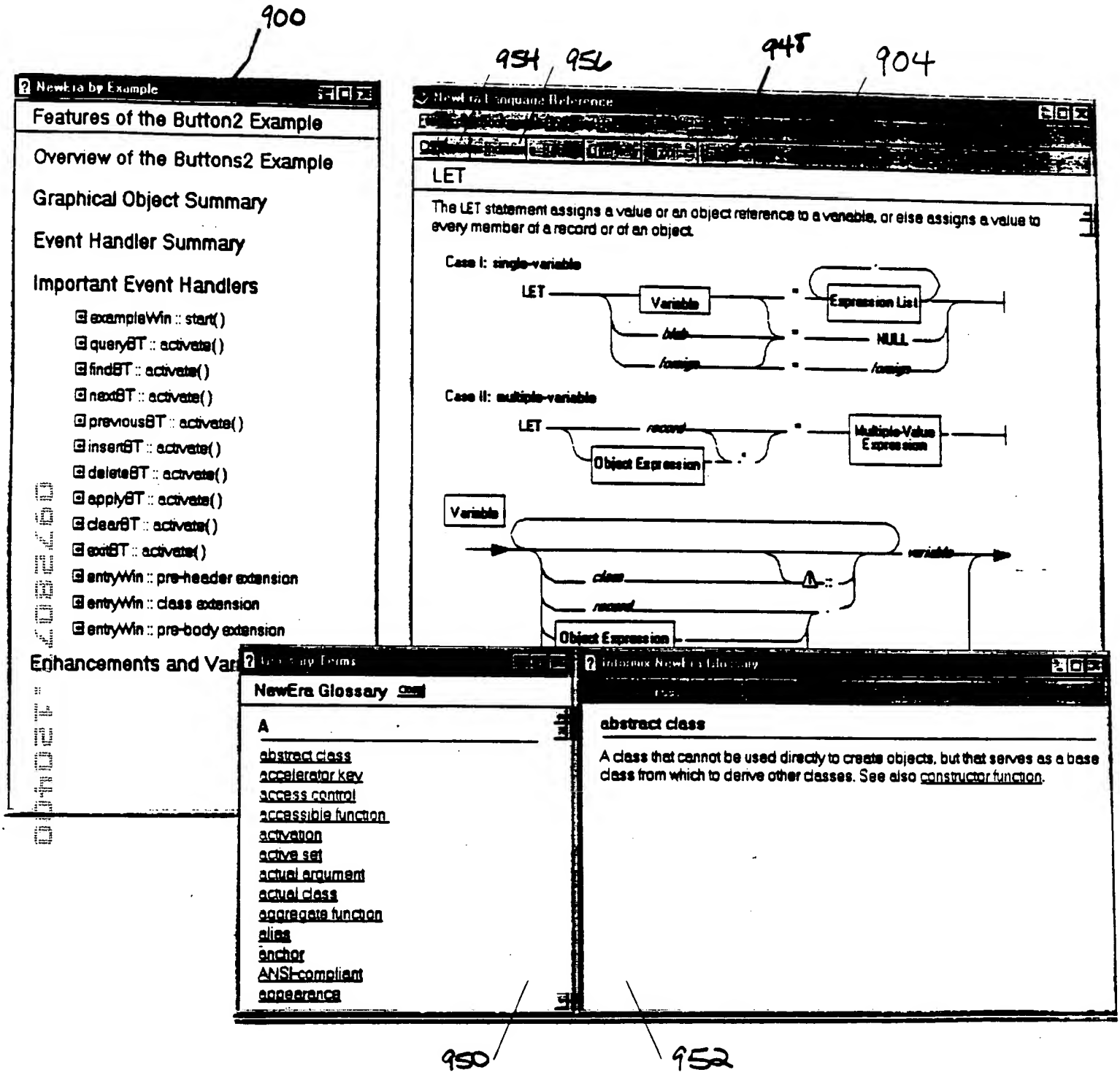

900
954
956
948
904
NewEra by Example
Features of the Button2 Example
Overview of the Buttons2 Example
Graphical Object Summary
Event Handler Summary
Important Event Handlers
exampleWin :: start()
queryBT :: activate()
findBT :: activate()
nextBT :: activate()
previousBT :: activate()
insertBT :: activate()
deleteBT :: activate()
applyBT :: activate()
clearBT :: activate()
exitBT :: activate()
entryWin :: pre-header extension
entryWin :: class extension
entryWin :: pre-body extension
Enhancements and Var
LET
The LET statement assigns a value or an object reference to a variable, or else assigns a value to every member of a record or of an object.
Case I: single-variable
LET
Variable
Expression List
NULL
Case II: multiple-variable
LET
Object Expression
Multiple-Value Expression
Variable
Object Expression
NewEra Glossary
A
abstract class
accelerator key
access control
accessible function
activation
active set
actual argument
actual class
aggregate function
alias
anchor
ANSI-compliant
appearance
abstract class
A class that cannot be used directly to create objects, but that serves as a base class from which to derive other classes. See also constructor function.
950
952

Fig. 9I

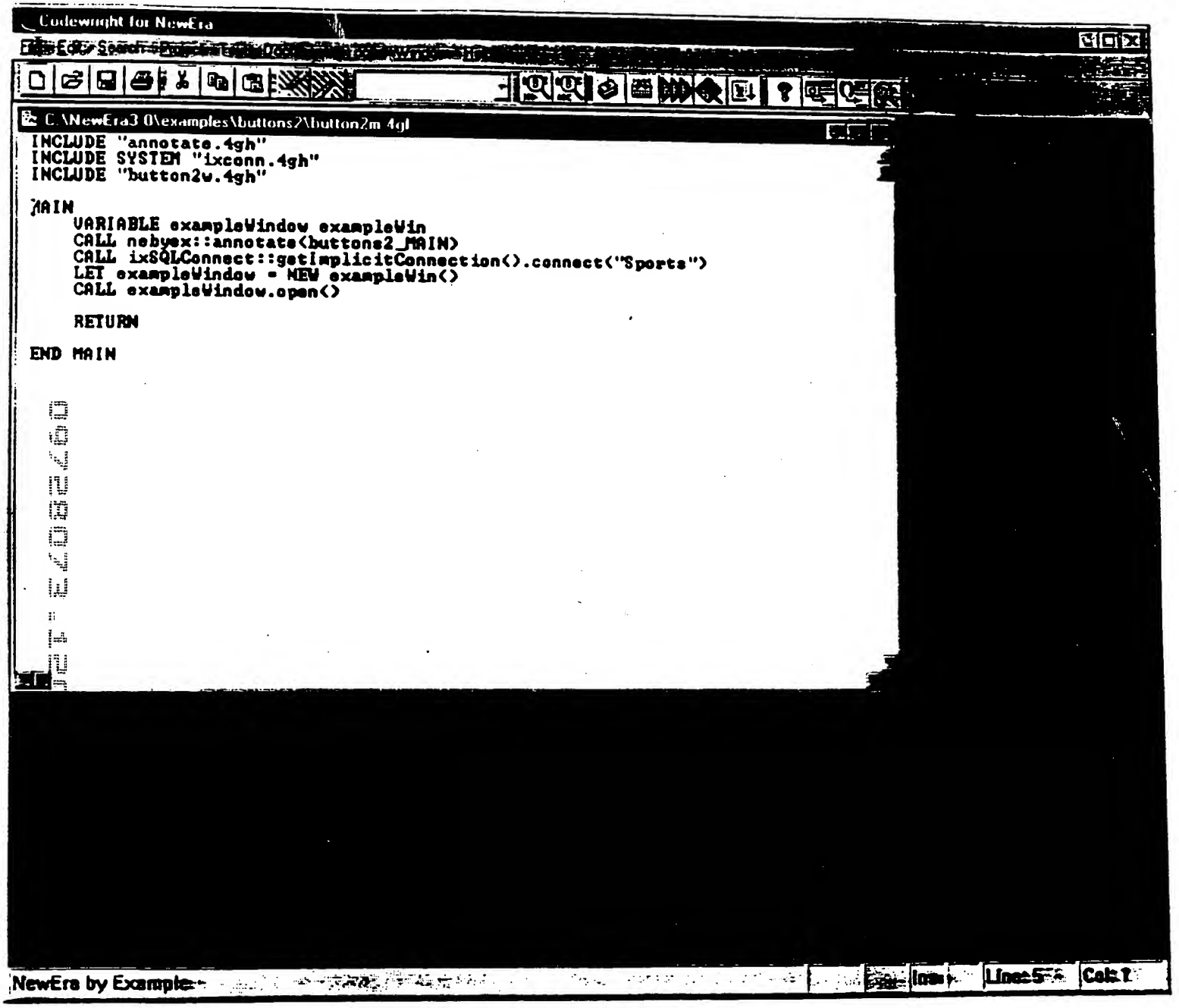

Codewright for NewEra
C:\NewEra3.0\examples\buttons2\button2m.4gl
INCLUDE "annotate.4gh"
INCLUDE SYSTEM "ixconn.4gh"
INCLUDE "button2w.4gh"
MAIN
VARIABLE exampleWindow exampleWin
CALL nebyex::annotate(buttons2_MAIN)
CALL ixSQLConnect::getImplicitConnection().connect("Sports")
LET exampleWindow = NEW exampleWin()
CALL exampleWindow.open()
RETURN
END MAIN
NewEra by Example
Lines:5
Col:1

Fig. 9J

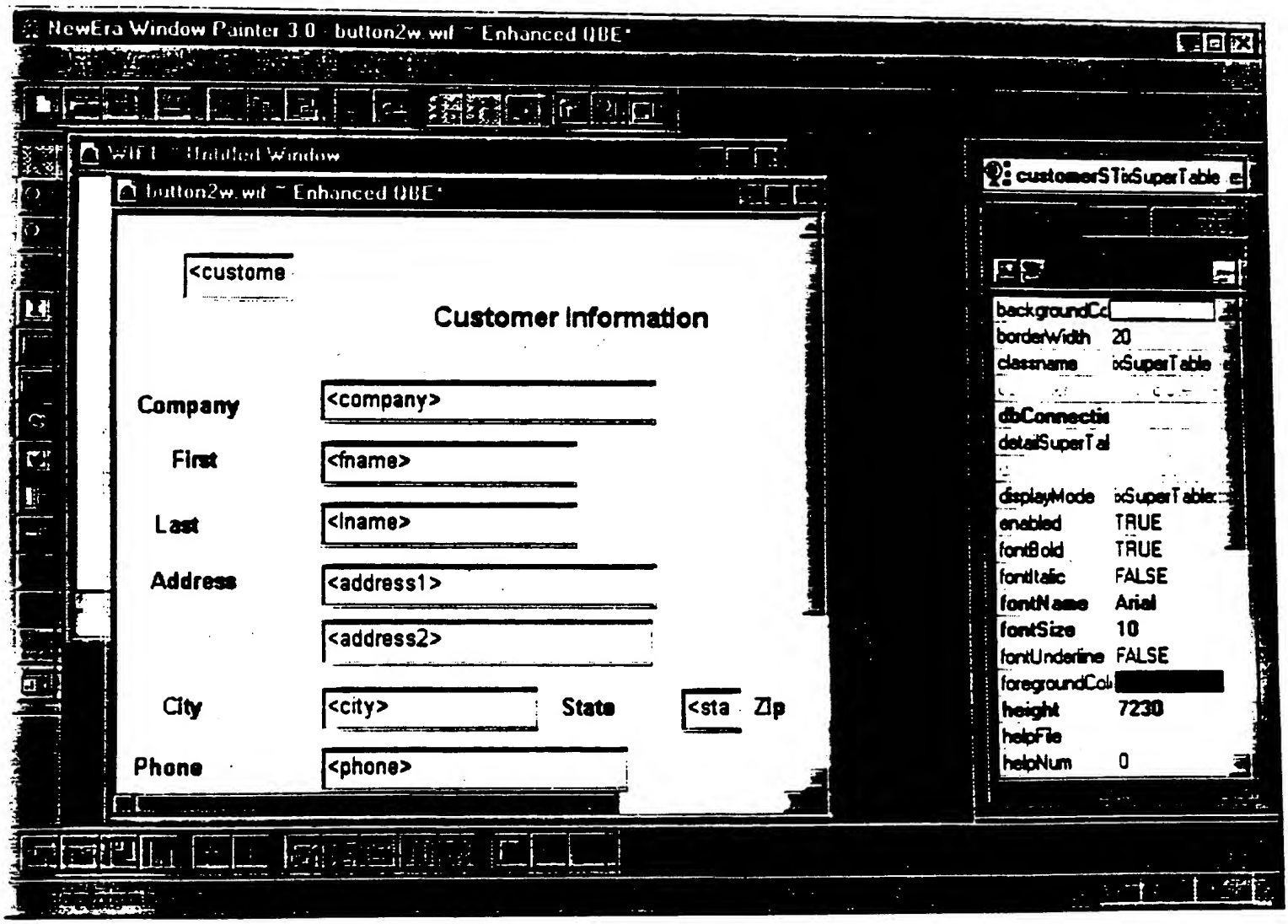

NewEra Window Painter 3.0 - button2w.wif ~ Enhanced QBE*
button2w.wif ~ Enhanced QBE*
<custome
Customer Information
Company
<company>
First
<fname>
Last
<lname>
Address
<address1>
<address2>
City
<city>
State
<sta
Zip
Phone
<phone>
customerSTtoSuperTable
backgroundCo
borderWidth 20
className ixSuperTable
dbConnectio
detailSuperTal
displayMode ixSuperTable
enabled TRUE
fontBold TRUE
fontItalic FALSE
fontName Arial
fontSize 10
fontUnderline FALSE
foregroundCol
height 7230
helpFile
helpNum 0

Fig. 4K

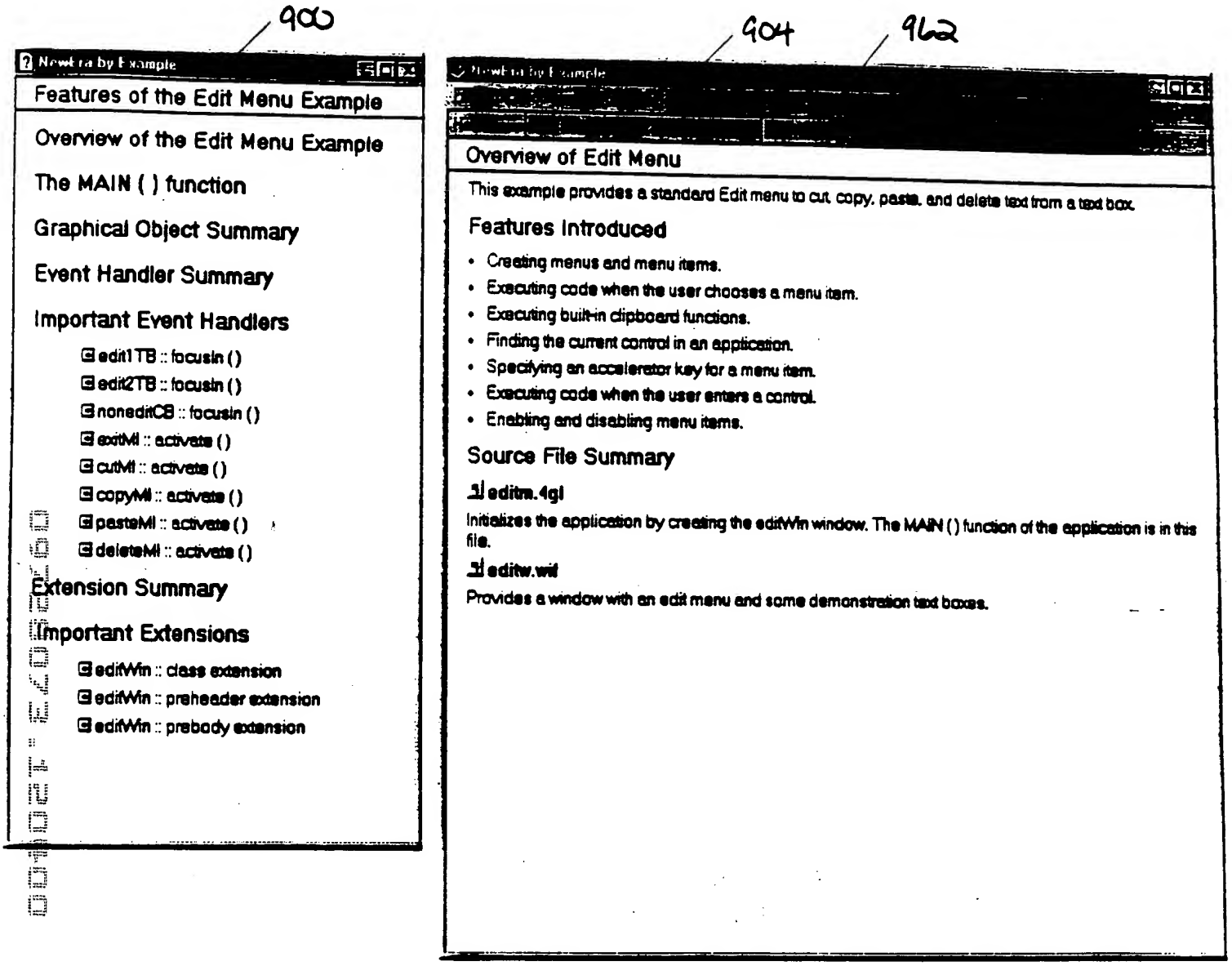

900
904
962
Features of the Edit Menu Example
Overview of the Edit Menu Example
The MAIN ( ) function
Graphical Object Summary
Event Handler Summary
Important Event Handlers
edit1TB :: focusIn ( )
edit2TB :: focusIn ( )
noneditCB :: focusIn ( )
exitMI :: activate ( )
cutMI :: activate ( )
copyMI :: activate ( )
pasteMI :: activate ( )
deleteMI :: activate ( )
Extension Summary
Important Extensions
editWin :: class extension
editWin :: preheader extension
editWin :: prebody extension
Overview of Edit Menu
This example provides a standard Edit menu to cut, copy, paste, and delete text from a text box.
Features Introduced
Creating menus and menu items.
Executing code when the user chooses a menu item.
Executing built-in clipboard functions.
Finding the current control in an application.
Specifying an accelerator key for a menu item.
Executing code when the user enters a control.
Enabling and disabling menu items.
Source File Summary
editw.4gl
Initializes the application by creating the editWin window. The MAIN ( ) function of the application is in this file.
editw.wil
Provides a window with an edit menu and some demonstration text boxes.

Fig. 9L

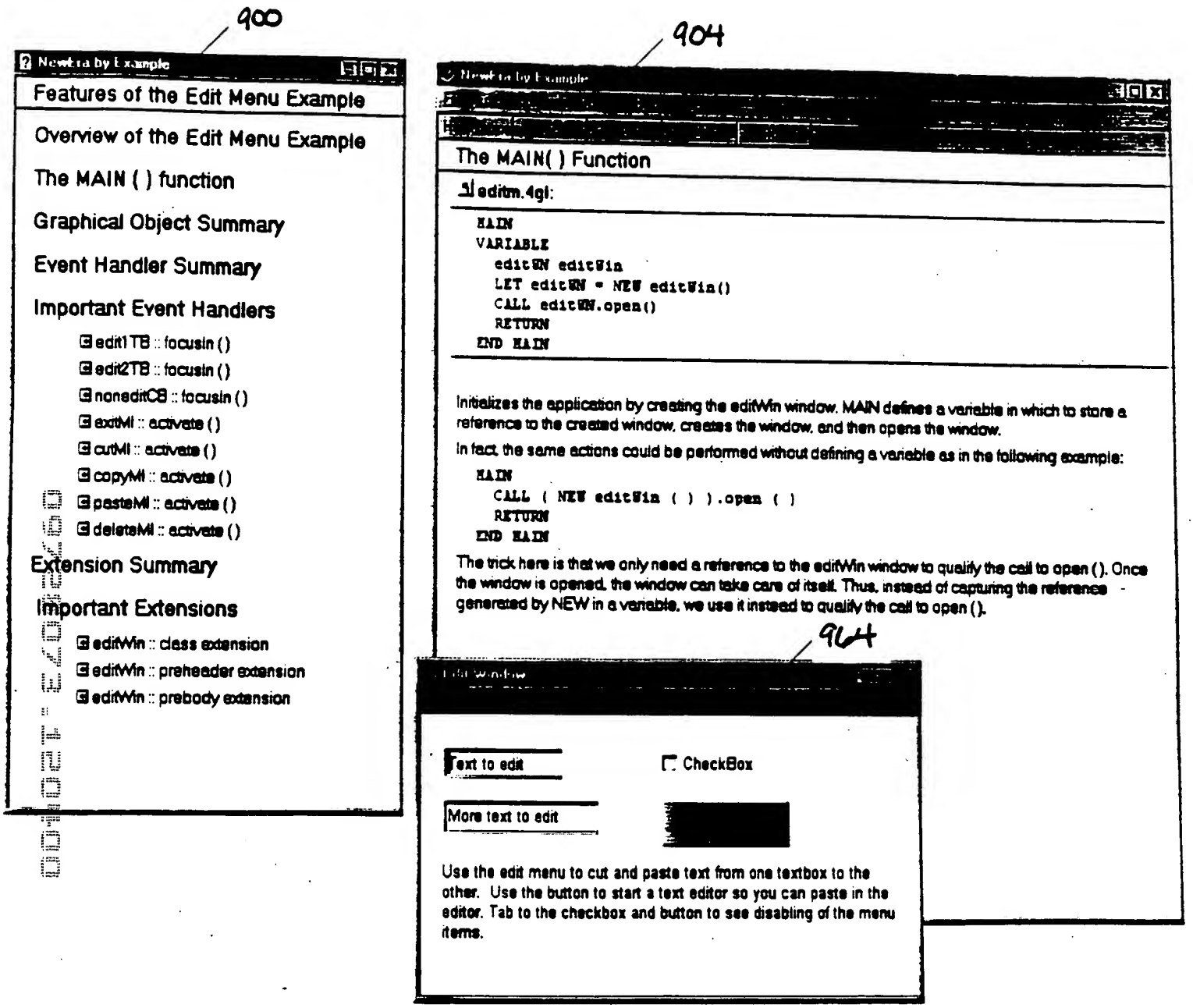

900
Newbie by Example
Features of the Edit Menu Example
Overview of the Edit Menu Example
The MAIN ( ) function
Graphical Object Summary
Event Handler Summary
Important Event Handlers
edit1TB :: focusIn ( )
edit2TB :: focusIn ( )
noneditCB :: focusIn ( )
exitMI :: activate ( )
cutMI :: activate ( )
copyMI :: activate ( )
pasteMI :: activate ( )
deleteMI :: activate ( )
Extension Summary
Important Extensions
editWin :: class extension
editWin :: preheader extension
editWin :: prebody extension
904
The MAIN( ) Function
editm.4gl:
MAIN
VARIABLE
editWN editWin
LET editWN = NEW editWin()
CALL editWN.open()
RETURN
END MAIN
Initializes the application by creating the editWin window. MAIN defines a variable in which to store a reference to the created window, creates the window, and then opens the window.
In fact, the same actions could be performed without defining a variable as in the following example:
MAIN
CALL (NEW editWin ( )).open ( )
RETURN
END MAIN
The trick here is that we only need a reference to the editWin window to qualify the call to open ( ). Once the window is opened, the window can take care of itself. Thus, instead of capturing the reference generated by NEW in a variable, we use it instead to qualify the call to open ( ).
964
Text to edit
CheckBox
More text to edit
Use the edit menu to cut and paste text from one textbox to the other. Use the button to start a text editor so you can paste in the editor. Tab to the checkbox and button to see disabling of the menu items.

Fig. 9M

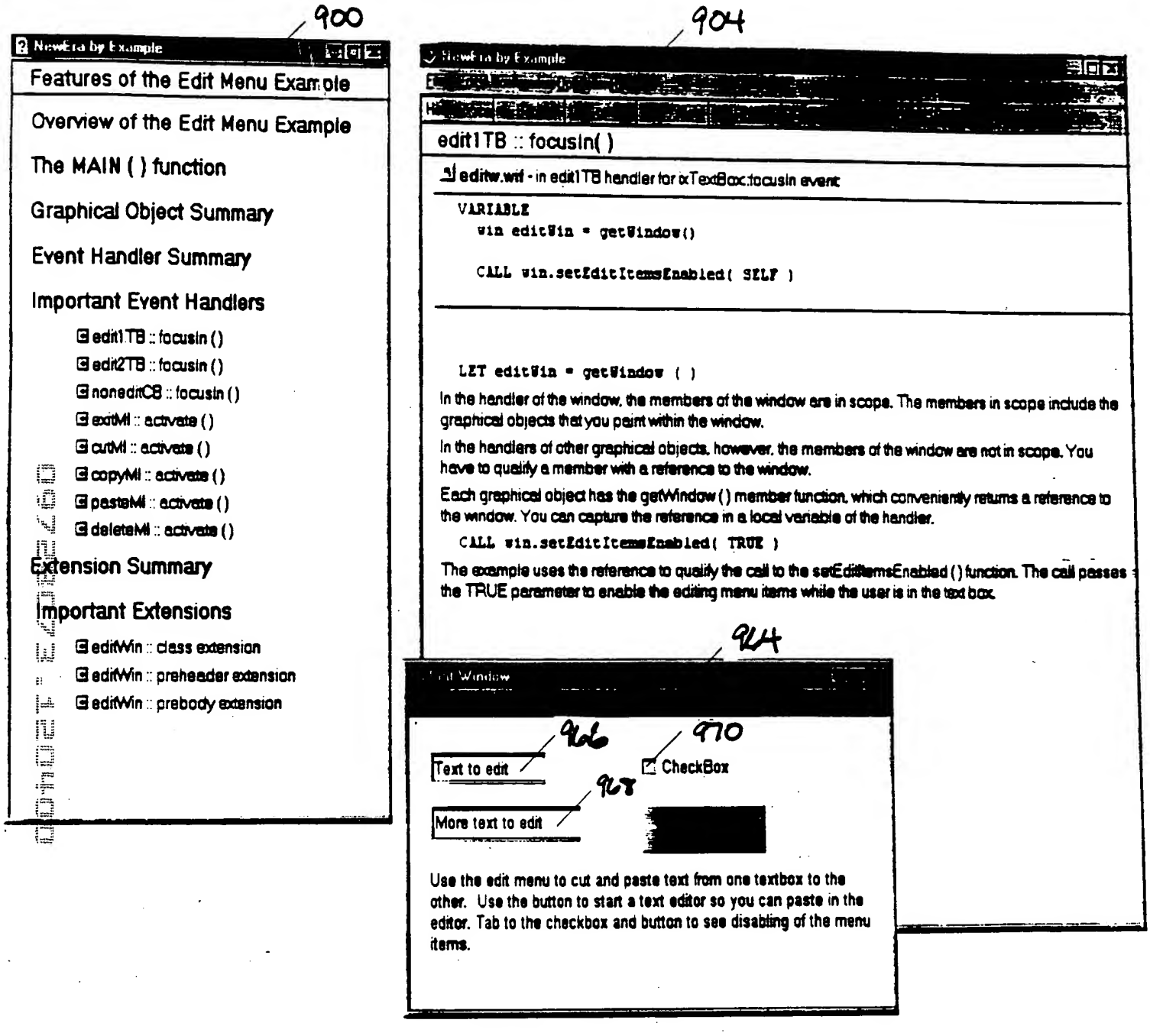

900
NewEra by Example
Features of the Edit Menu Example
Overview of the Edit Menu Example
The MAIN ( ) function
Graphical Object Summary
Event Handler Summary
Important Event Handlers
edit1TB :: focusIn ( )
edit2TB :: focusIn ( )
noneditCB :: focusIn ( )
exitMI :: activate ( )
cutMI :: activate ( )
copyMI :: activate ( )
pasteMI :: activate ( )
deleteMI :: activate ( )
Extension Summary
Important Extensions
editWin :: class extension
editWin :: preheader extension
editWin :: prebody extension
904
NewEra by Example
edit1TB :: focusIn( )
editw.wfl - in edit1TB handler for ixTextBox:focusIn event
VARIABLE
win editWin = getWindow()
CALL win.setEditItemsEnabled(SELF)
LET editWin = getWindow ( )
In the handler of the window, the members of the window are in scope. The members in scope include the graphical objects that you paint within the window.
In the handlers of other graphical objects, however, the members of the window are not in scope. You have to qualify a member with a reference to the window.
Each graphical object has the getWindow ( ) member function, which conveniently returns a reference to the window. You can capture the reference in a local variable of the handler.
CALL win.setEditItemsEnabled(TRUE)
The example uses the reference to qualify the call to the setEditItemsEnabled ( ) function. The call passes the TRUE parameter to enable the editing menu items while the user is in the text box.
914
Edit Window
926
970
Text to edit
CheckBox
968
More text to edit
Use the edit menu to cut and paste text from one textbox to the other. Use the button to start a text editor so you can paste in the editor. Tab to the checkbox and button to see disabling of the menu items.

Fig. 9N

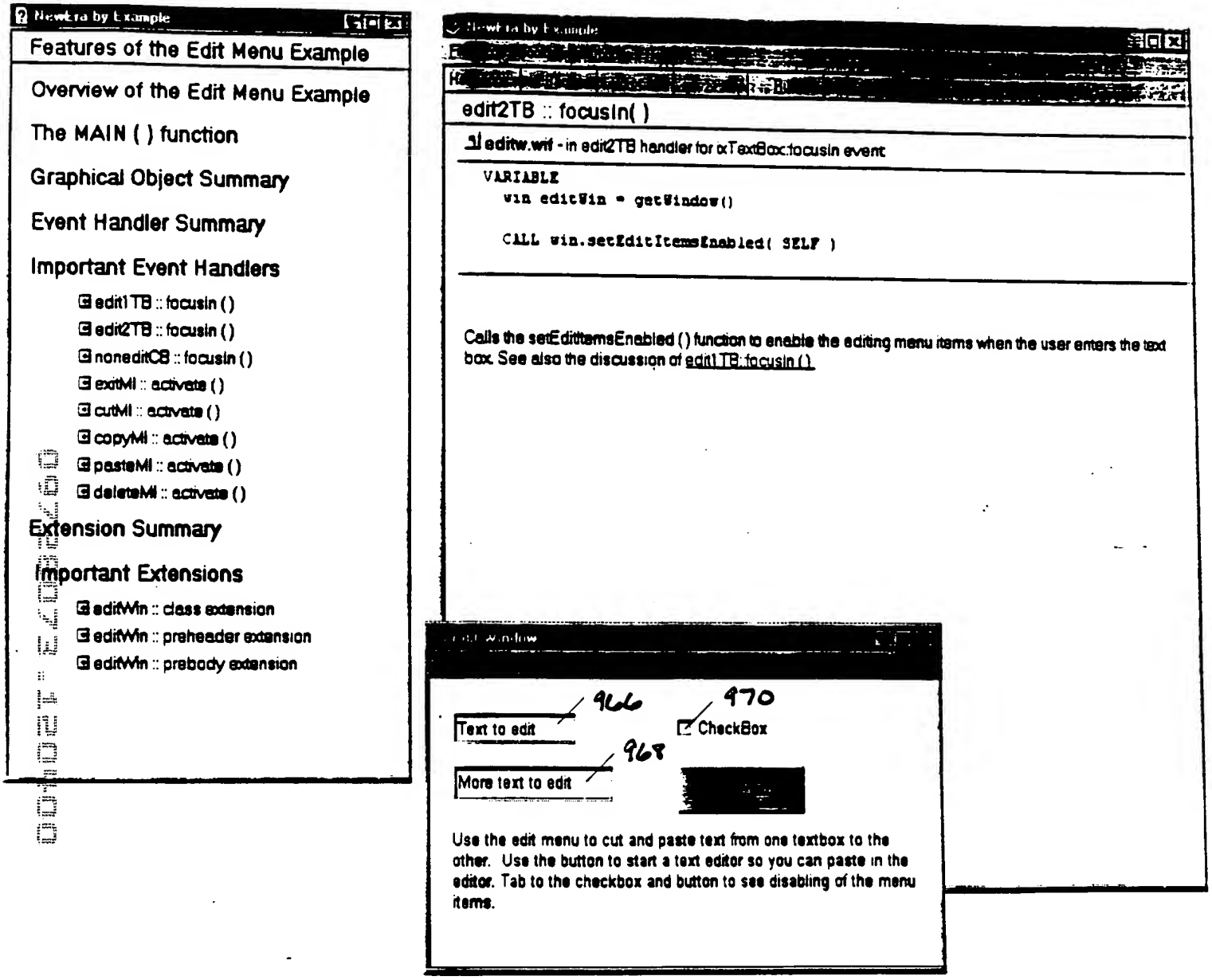

Features of the Edit Menu Example
Overview of the Edit Menu Example
The MAIN ( ) function
Graphical Object Summary
Event Handler Summary
Important Event Handlers
edit1TB :: focusIn ( )
edit2TB :: focusIn ( )
noneditCB :: focusIn ( )
exitMI :: activate ( )
cutMI :: activate ( )
copyMI :: activate ( )
pasteMI :: activate ( )
deleteMI :: activate ( )
Extension Summary
Important Extensions
editWin :: class extension
editWin :: preheader extension
editWin :: prebody extension
edit2TB :: focusIn( )
editw.wif - in edit2TB handler for txTextBox:focusIn event
VARIABLE
win editWin = getWindow()
CALL win.setEditItemsEnabled(SELF)
Calls the setEditItemsEnabled ( ) function to enable the editing menu items when the user enters the text box. See also the discussion of edit1TB::focusIn ( )
966
970
Text to edit
CheckBox
968
More text to edit
Use the edit menu to cut and paste text from one textbox to the other. Use the button to start a text editor so you can paste in the editor. Tab to the checkbox and button to see disabling of the menu items.

Fig. 90

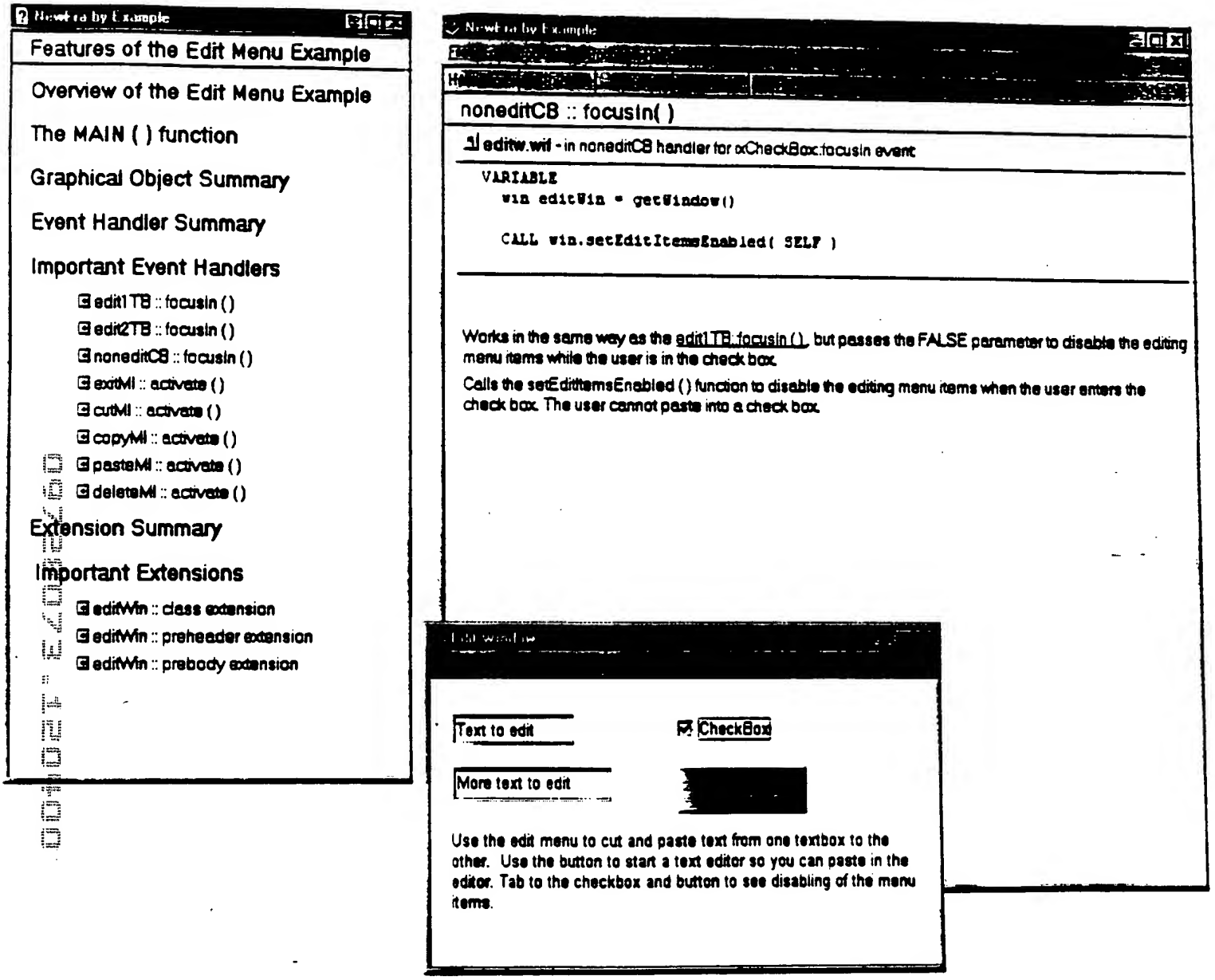

Features of the Edit Menu Example
Overview of the Edit Menu Example
The MAIN ( ) function
Graphical Object Summary
Event Handler Summary
Important Event Handlers
edit1TB :: focusIn ( )
edit2TB :: focusIn ( )
noneditCB :: focusIn ( )
exitMI :: activate ( )
cutMI :: activate ( )
copyMI :: activate ( )
pasteMI :: activate ( )
deleteMI :: activate ( )
Extension Summary
Important Extensions
editWin :: class extension
editWin :: preheader extension
editWin :: prebody extension
noneditCB :: focusIn( )
editw.wil - in noneditCB handler for ctCheckBox:focusIn event
VARIABLE
win editWin = getWindow()
CALL win.setEditItemsEnabled(SELF)
Works in the same way as the edit1TB:focusIn ( ), but passes the FALSE parameter to disable the editing menu items while the user is in the check box.
Calls the setEditItemsEnabled ( ) function to disable the editing menu items when the user enters the check box. The user cannot paste into a check box.
Text to edit
CheckBox
More text to edit
Use the edit menu to cut and paste text from one textbox to the other. Use the button to start a text editor so you can paste in the editor. Tab to the checkbox and button to see disabling of the menu items.

Fig. 9P

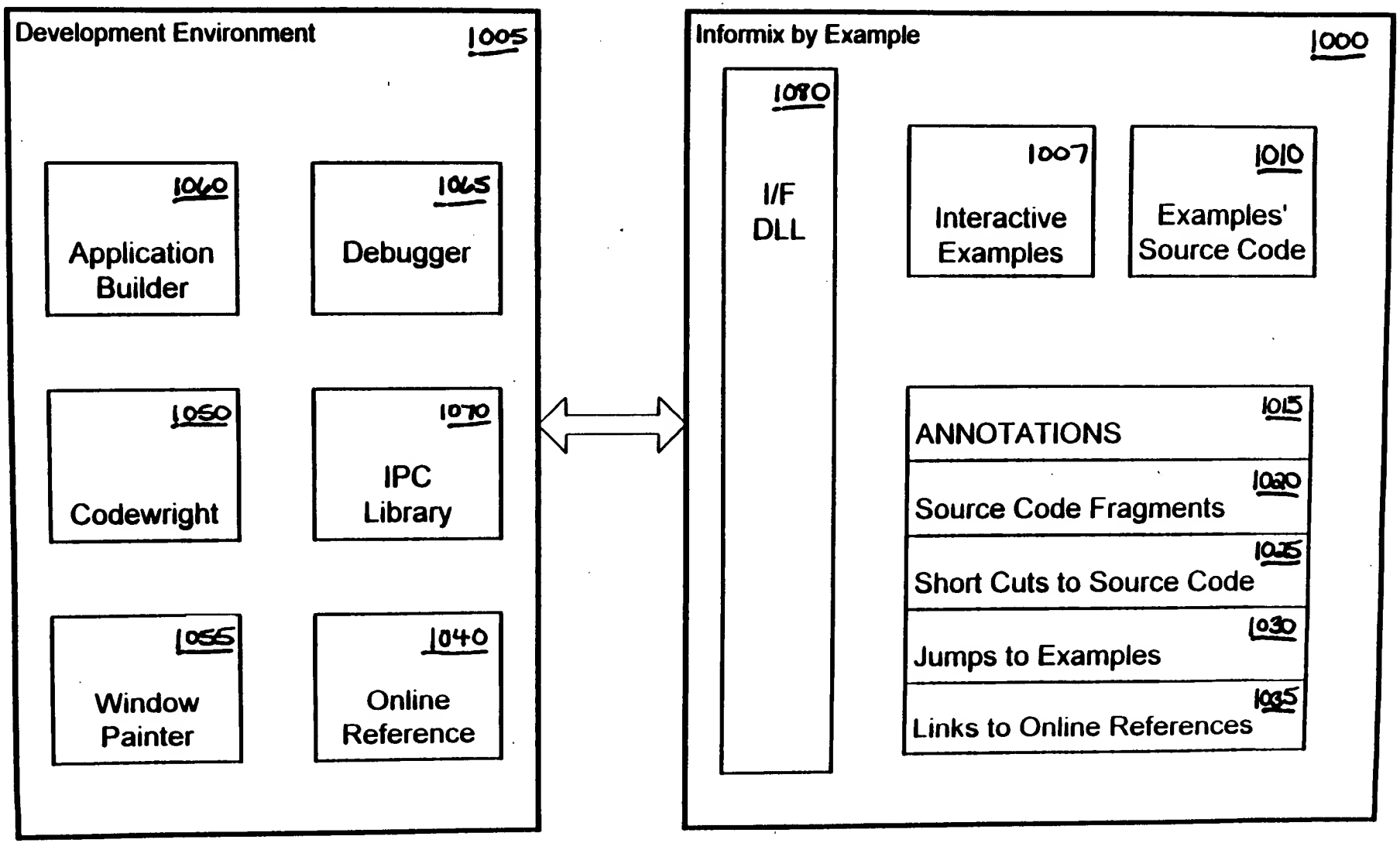


Fig. 10

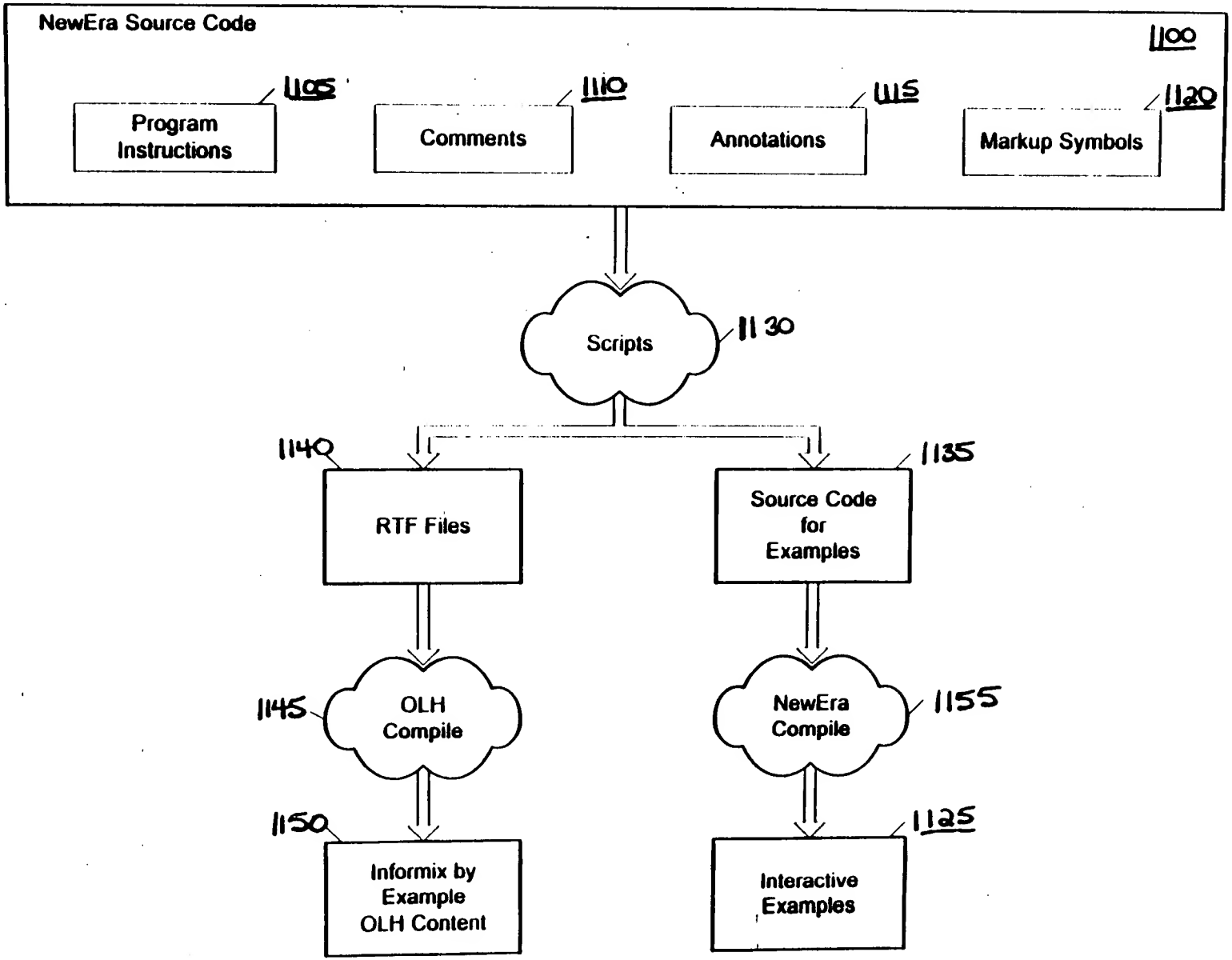

NewEra Source Code
1100
Program Instructions
1105
Comments
1110
Annotations
1115
Markup Symbols
1120
Scripts
1130
RTF Files
1140
Source Code for Examples
1135
OLH Compile
1145
NewEra Compile
1155
Informix by Example OLH Content
1150
Interactive Examples
1125

Fig. 11

```
FUNCTION driveStockRpt( destType SMALLINT, destName CHAR(*) ) RETUR
NING VOID
{.normal
Since objects, in particular ixRow objects, cannot be passed
as arguments to the report formatter, rows of fetched data will
be unpacked into a record that matches the data types and lengths
of elements in the fetched rows.
}
VARIABLE
     stockRec RECORD
          mn CHAR(15),     -- manufact.manu_name
          sn SMALLINT,     -- stock.stock_num
          sd CHAR(15),     -- stock.description
          sp MONEY(6,2),   -- stock.unit_price
          su CHAR(4)       -- stock.unit
          END RECORD,

     stockStmt ixSQLStmt,
     stmtString CHAR(*),
     stockRow ixRow,

     errorCode INTEGER,
     logFile ixErrorLog

{.normal
Use the implicit connection object to create an SQL statement
object. The connection object must already be connected to a
database.
Checking the status of the prepare( ) call will confirm this.
}
{.[edit stmt}
LET stockStmt =
ixSQLConnect::getImplicitConnection().createStmtObject()
{.]file stmt}
```

1200 1205 1210 1215

Fig. 12

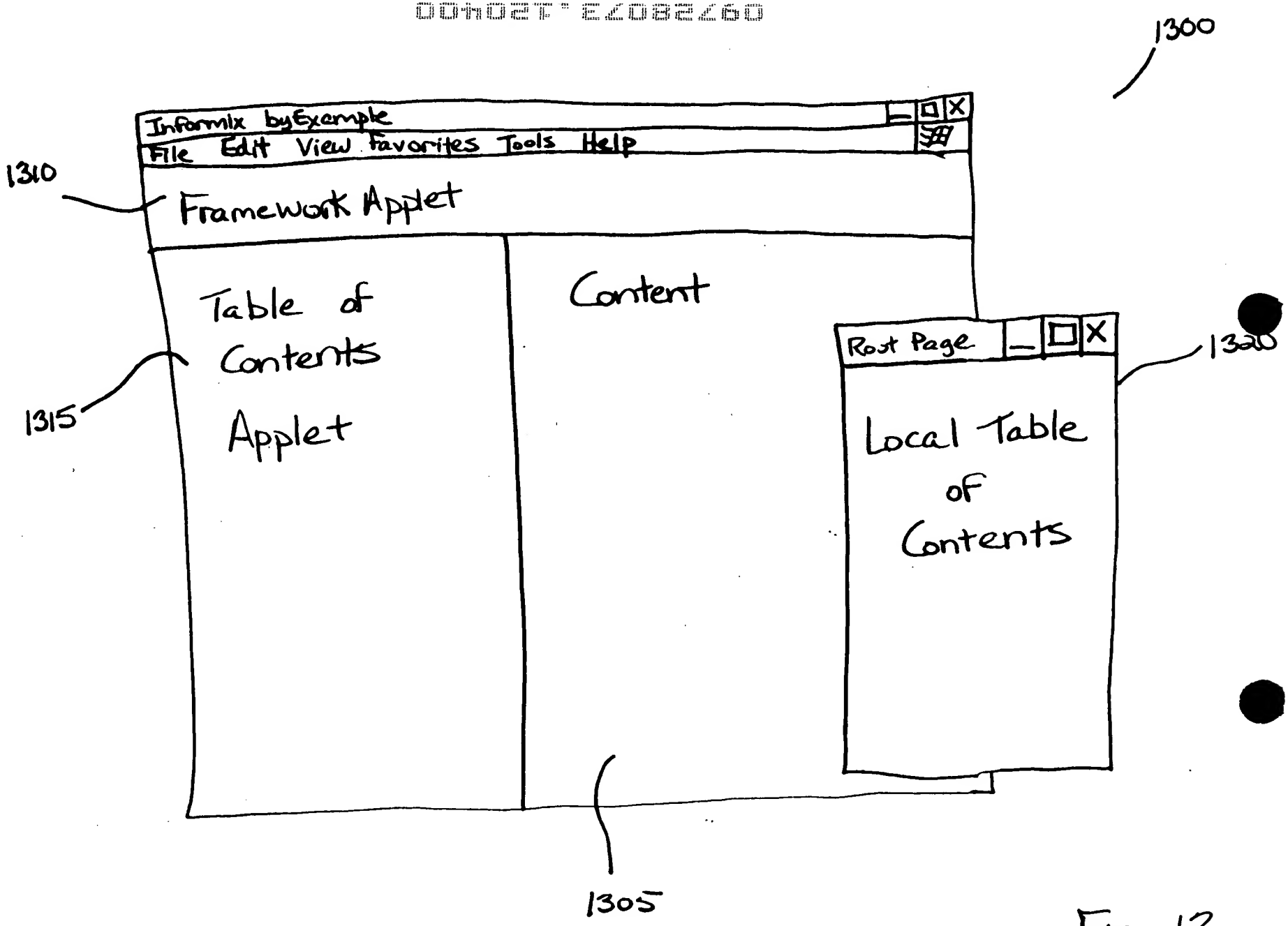


Fig. 13

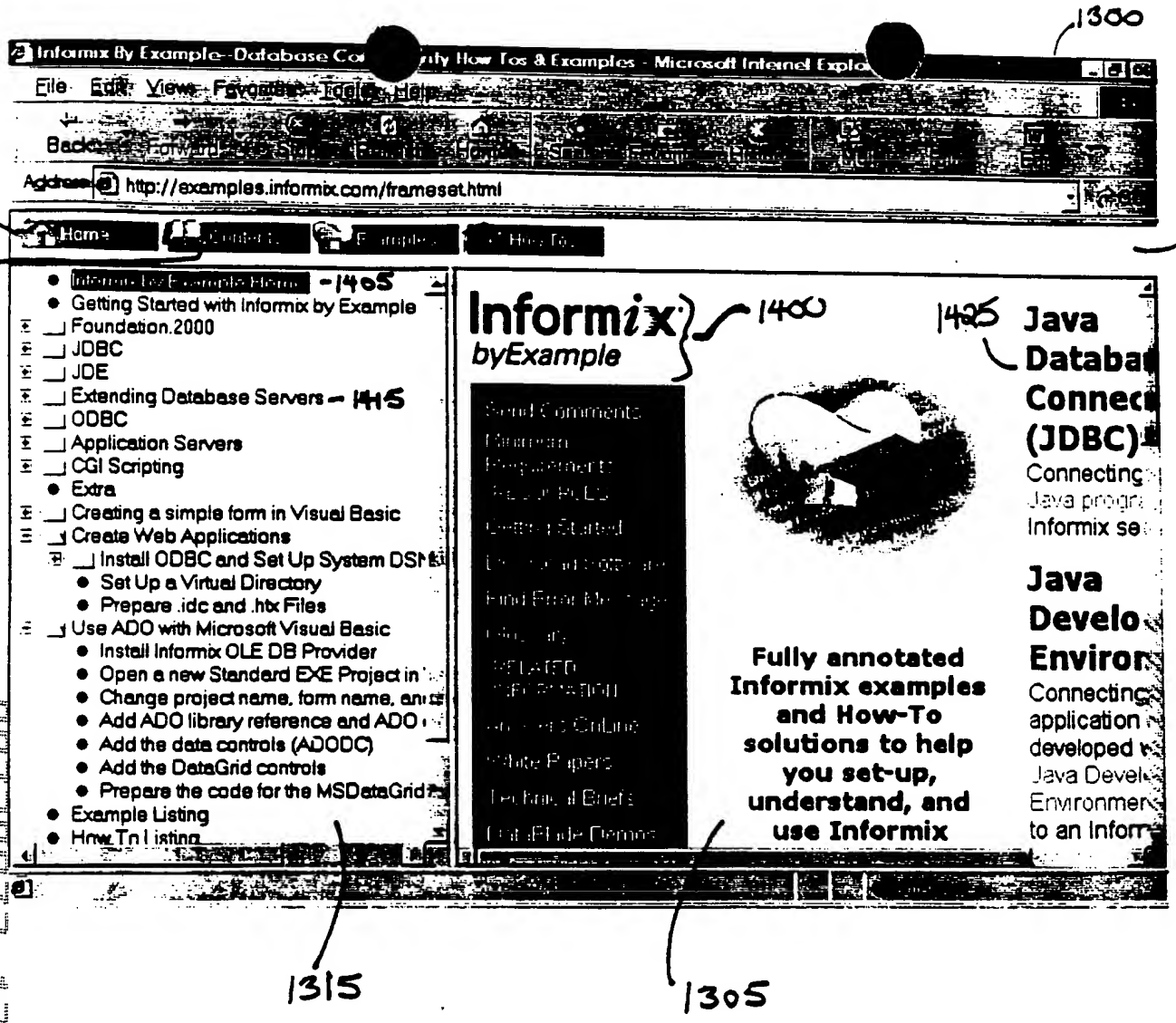

1300
Informix By Example--Database Co
ly How Tos & Examples - Microsoft Internet Explo
File Edit View Favorites Tools Help
Back Forward
Address http://examples.informix.com/frameset.html
1420
Home
1310
1410
Informix By Example Home
1405
Getting Started with Informix by Example
Foundation.2000
JDBC
JDE
Extending Database Servers
1415
ODBC
Application Servers
CGI Scripting
Extra
Creating a simple form in Visual Basic
Create Web Applications
Install ODBC and Set Up System DSN
Set Up a Virtual Directory
Prepare .idc and .htx Files
Use ADO with Microsoft Visual Basic
Install Informix OLE DB Provider
Open a new Standard EXE Project in
Change project name, form name, an
Add ADO library reference and ADO
Add the data controls (ADODC)
Add the DataGrid controls
Prepare the code for the MSDataGrid
Example Listing
How To Listing
Informix
byExample
1400
1425
Java
Databa
Connec
(JDBC)
Connecting
Java progra
Informix se
Java
Develo
Environ
Connecting
application
developed
Java Devel
Environmen
to an Inform
Fully annotated Informix examples and How-To solutions to help you set-up, understand, and use Informix
1315
1305

Fig. 14A

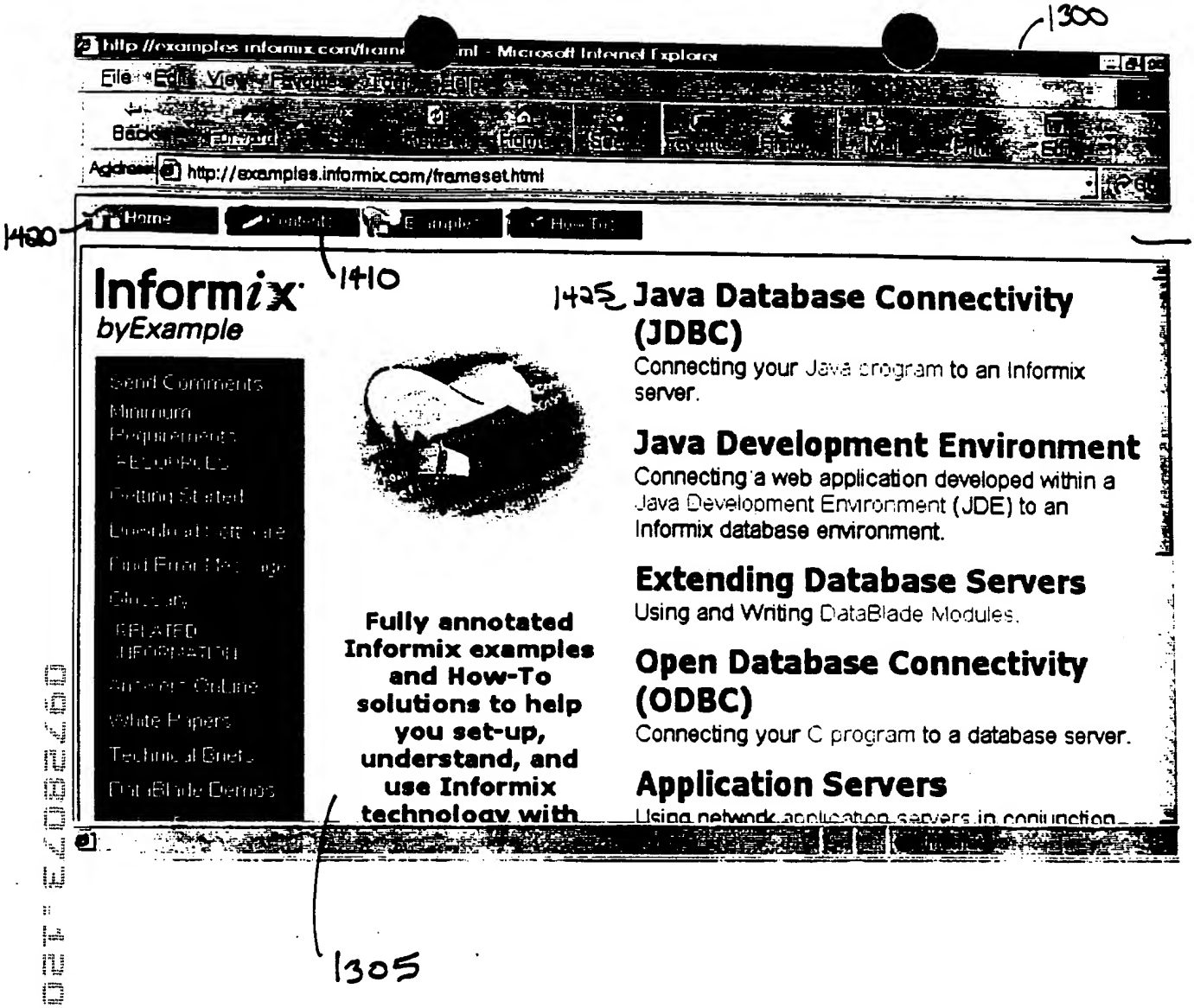

1300
http://examples.informix.com/frameset.html
Home
1420
1310
1410
Informix
byExample
1425
Java Database Connectivity (JDBC)
Connecting your Java program to an Informix server.
Java Development Environment
Connecting a web application developed within a Java Development Environment (JDE) to an Informix database environment.
Extending Database Servers
Using and Writing DataBlade Modules.
Open Database Connectivity (ODBC)
Connecting your C program to a database server.
Application Servers
Send Comments
Minimum Requirements
Getting Started
Online Publications
White Papers
Technical Briefs
DataBlade Demos
Fully annotated Informix examples and How-To solutions to help you set-up, understand, and use Informix technology with
1305

Fig. 14B

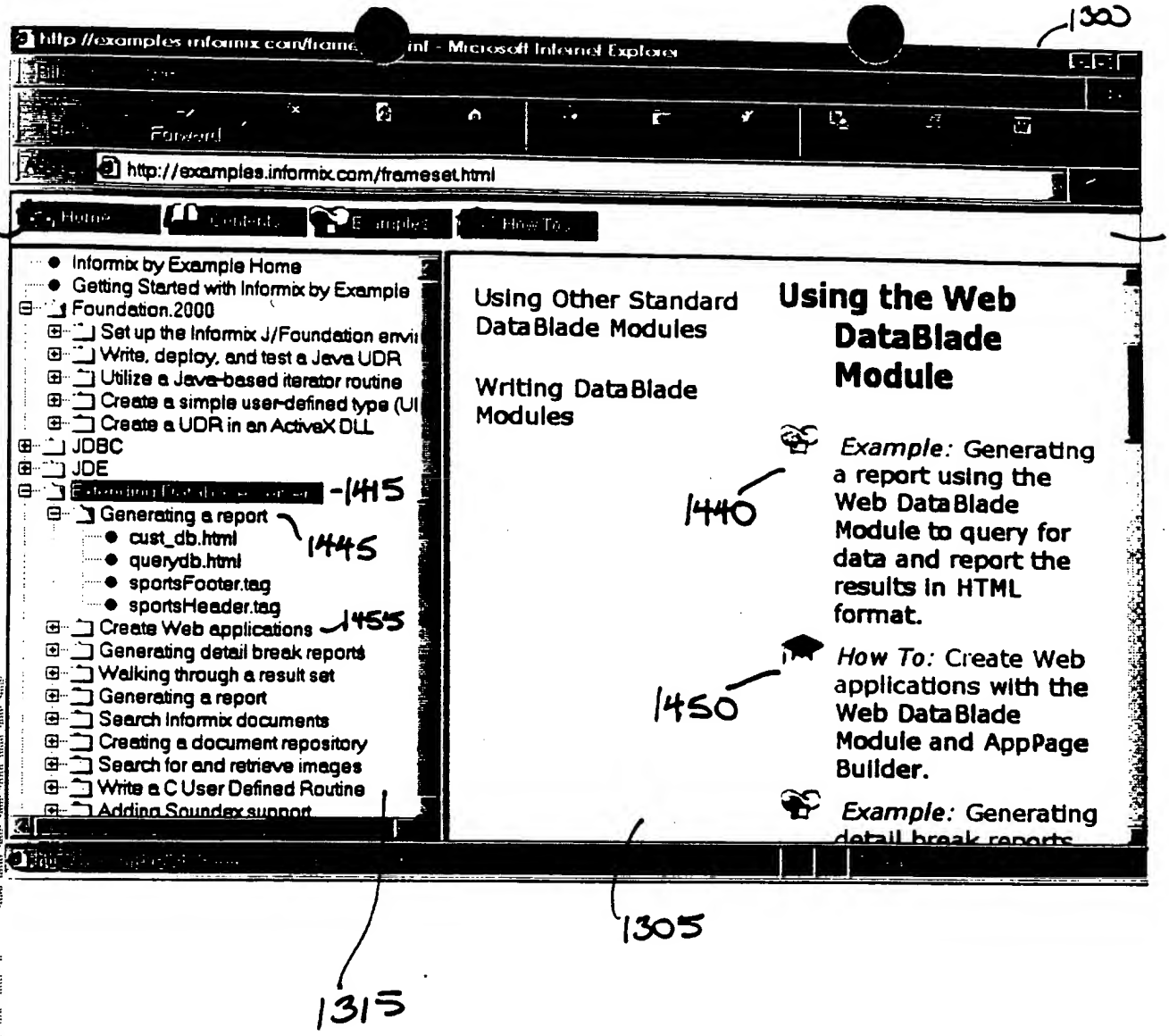

1300
http://examples.informix.com/frameset.html
1420
1310
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
Set up the Informix J/Foundation envi
Write, deploy, and test a Java UDR
Utilize a Java-based iterator routine
Create a simple user-defined type (UI
Create a UDR in an ActiveX DLL
JDBC
JDE
1415
Generating a report
cust_db.html
querydb.html
sportsFooter.tag
sportsHeader.tag
1445
1455
Create Web applications
Generating detail break reports
Walking through a result set
Generating a report
Search Informix documents
Creating a document repository
Search for and retrieve images
Write a C User Defined Routine
Adding Soundex support
Using Other Standard DataBlade Modules
Writing DataBlade Modules
Using the Web DataBlade Module
1440
Example: Generating a report using the Web DataBlade Module to query for data and report the results in HTML format.
1450
How To: Create Web applications with the Web DataBlade Module and AppPage Builder.
Example: Generating
1305
1315

Fig. 14C

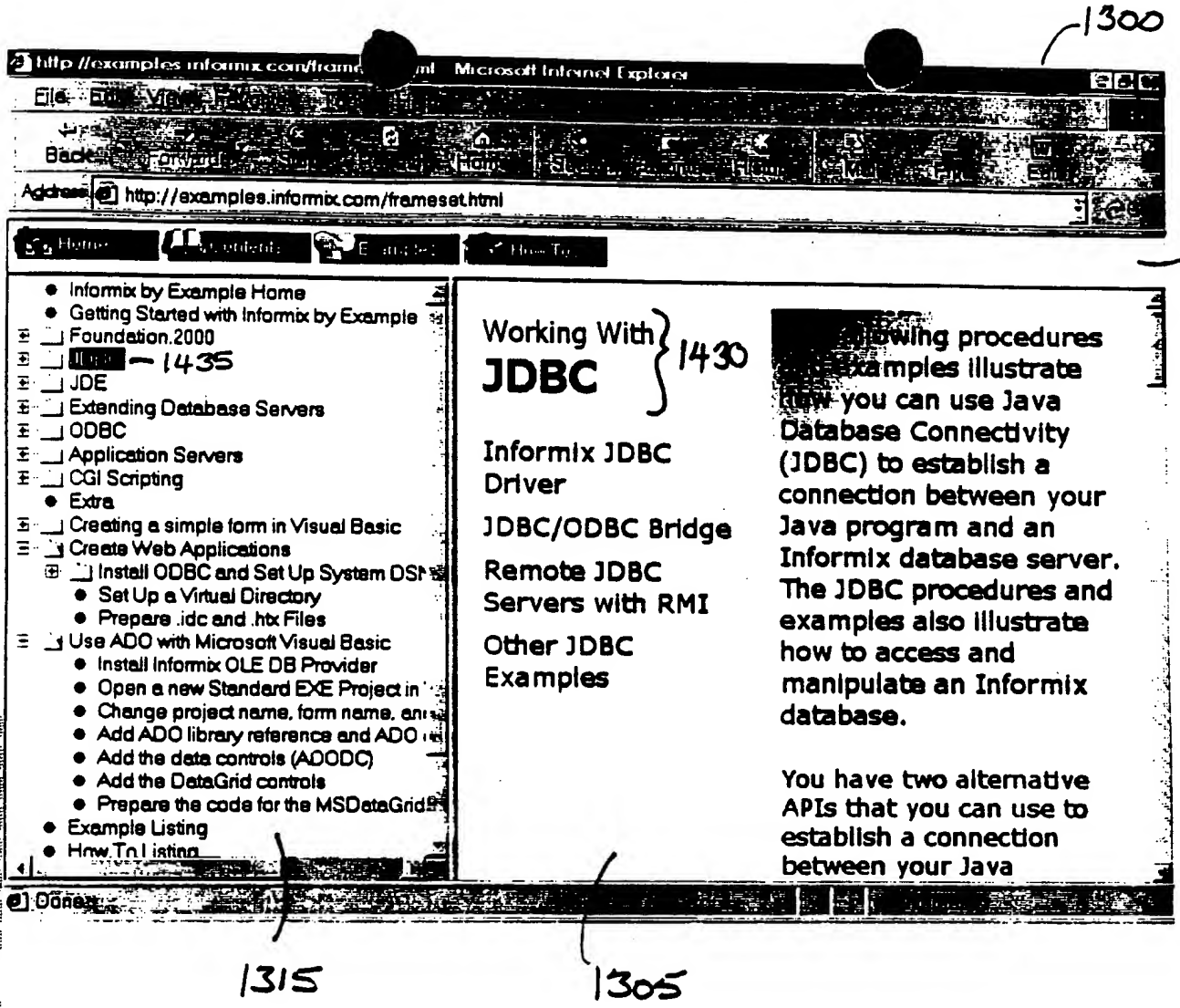

1300
http://examples.informix.com/frameset.html
1310
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
1435
JDE
Extending Database Servers
ODBC
Application Servers
CGI Scripting
Extra
Creating a simple form in Visual Basic
Create Web Applications
Install ODBC and Set Up System DSN
Set Up a Virtual Directory
Prepare .idc and .htx Files
Use ADO with Microsoft Visual Basic
Install Informix OLE DB Provider
Open a new Standard EXE Project in
Change project name, form name, and
Add ADO library reference and ADO
Add the data controls (ADODC)
Add the DataGrid controls
Prepare the code for the MSDataGrid
Example Listing
Working With
JDBC
1430
Informix JDBC Driver
JDBC/ODBC Bridge
Remote JDBC Servers with RMI
Other JDBC Examples
following procedures
examples illustrate
how you can use Java Database Connectivity (JDBC) to establish a connection between your Java program and an Informix database server. The JDBC procedures and examples also illustrate how to access and manipulate an Informix database.
You have two alternative APIs that you can use to establish a connection between your Java
1315
1305

Fig. 14D

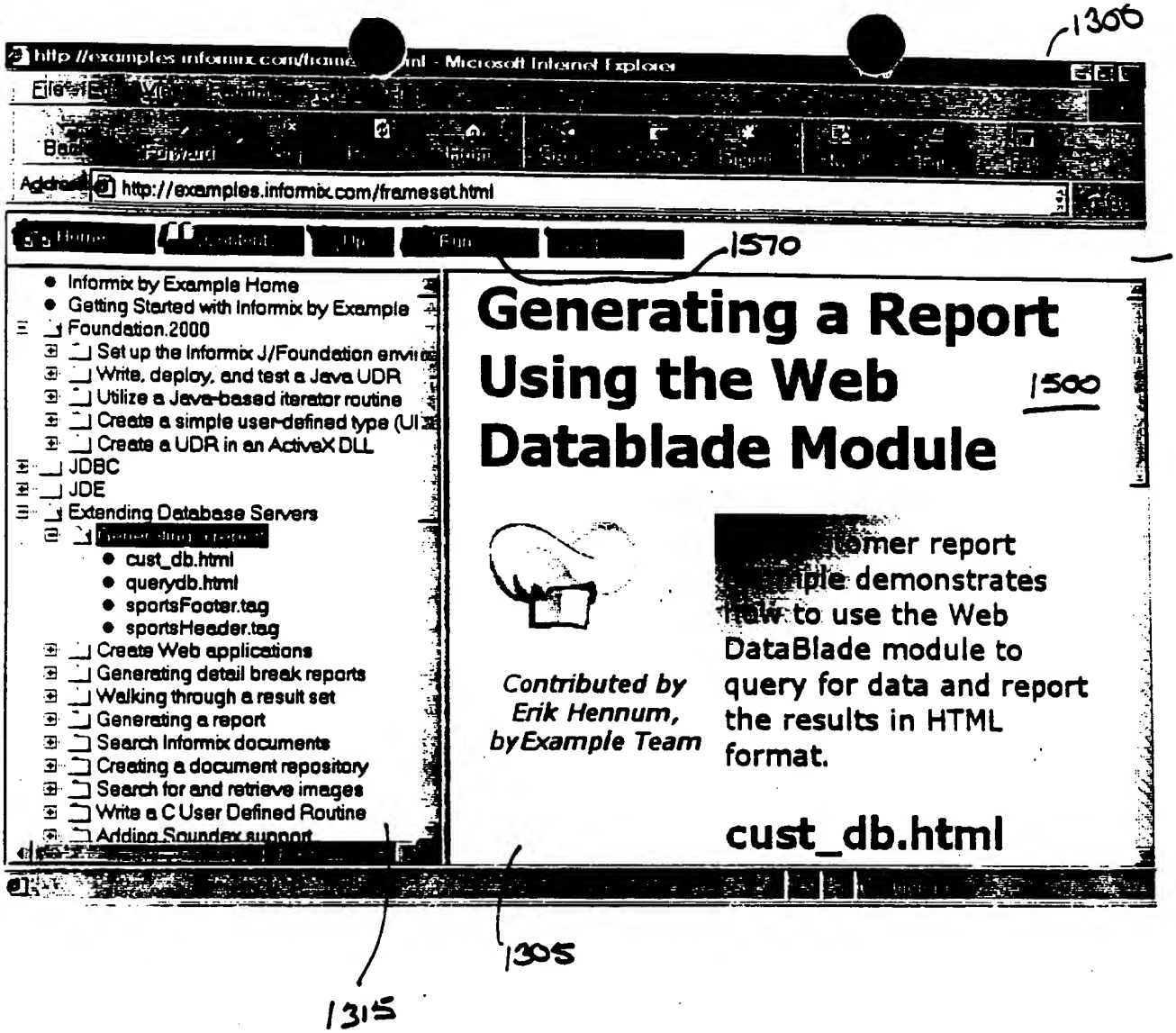

1300
http://examples.informix.com/frameset.html
1570
1310
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
Set up the Informix J/Foundation envi
Write, deploy, and test a Java UDR
Utilize a Java-based iterator routine
Create a simple user-defined type (UI
Create a UDR in an ActiveX DLL
JDBC
JDE
Extending Database Servers
cust_db.html
querydb.html
sportsFooter.tag
sportsHeader.tag
Create Web applications
Generating detail break reports
Walking through a result set
Generating a report
Search Informix documents
Creating a document repository
Search for and retrieve images
Write a C User Defined Routine
Adding Soundex support
Generating a Report Using the Web Datablade Module
1500
Contributed by Erik Hennum, byExample Team
omer report ple demonstrates w to use the Web DataBlade module to query for data and report the results in HTML format.
cust_db.html
1305
1315

Fig. 15A

Generating a Report Using the Web Datablade Module

1500

Contributed by Erik Hennum, byExample Team

The customer report example demonstrates how to use the Web DataBlade module to query for data and report the results in HTML format.

1505

cust_db.html – 1510

1505

This app page accepts a query and generates an HTML report

querydb.html – 1510

1505

This HTML page contains a form that invokes an app page

sportsFooter.tag – 1510

1505

The sportsFooter dynamic tag generates the footer for an app page.

sportsHeader.tag – 1510

1505

The sportsHeader dynamic tag generates the header for an app page.

Click here to view or print all of the source files for this example.

Fig. 15B

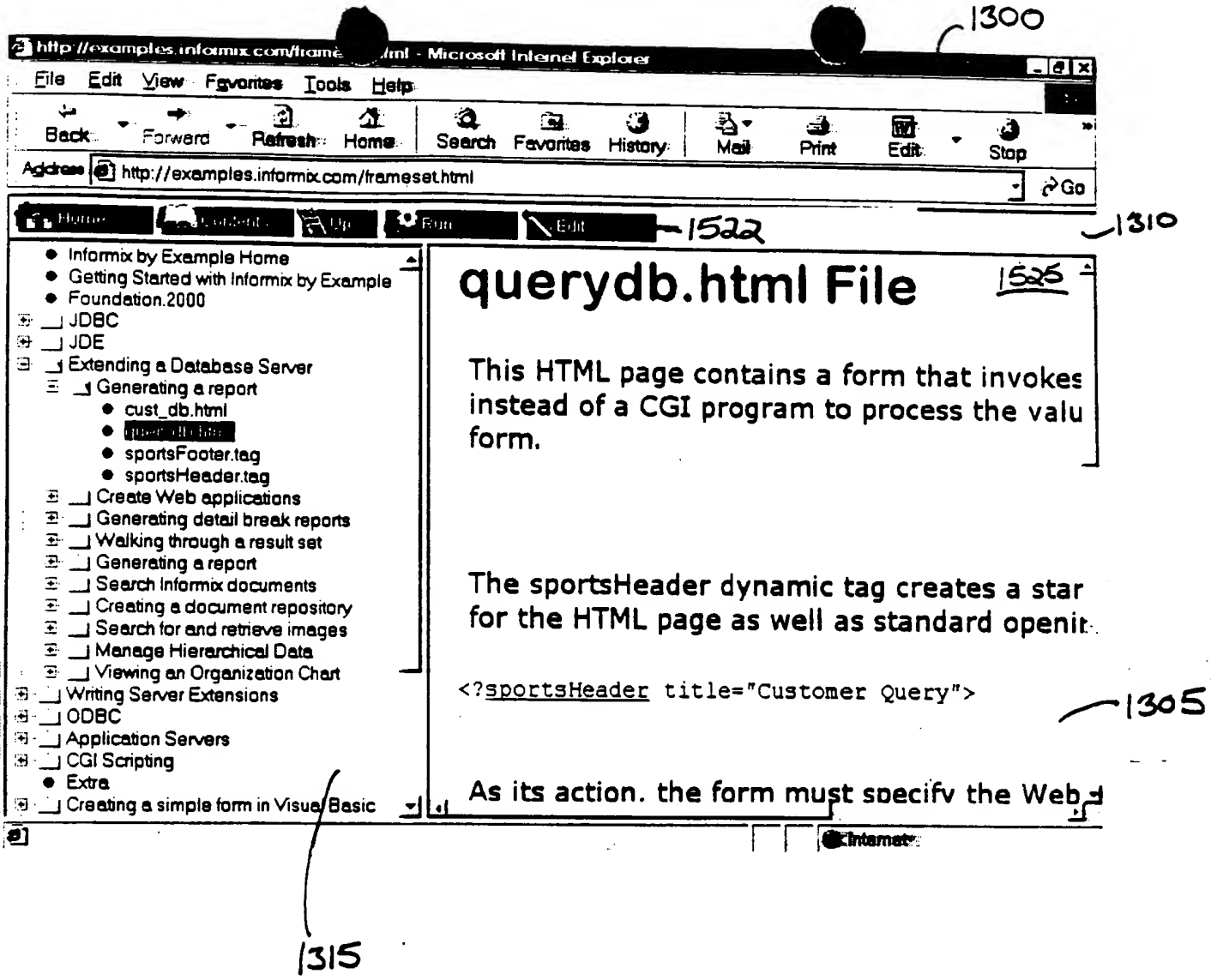

1300
http://examples.informix.com/frame .html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Refresh Home Search Favorites History Mail Print Edit Stop
Address http://examples.informix.com/frameset.html
Go
Home Contents Up Run Edit
1522
1310
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
JDBC
JDE
Extending a Database Server
Generating a report
cust_db.html
querydb.html
sportsFooter.tag
sportsHeader.tag
Create Web applications
Generating detail break reports
Walking through a result set
Generating a report
Search Informix documents
Creating a document repository
Search for and retrieve images
Manage Hierarchical Data
Viewing an Organization Chart
Writing Server Extensions
ODBC
Application Servers
CGI Scripting
Extra
Creating a simple form in Visual Basic
1315
querydb.html File
1525
This HTML page contains a form that invokes
instead of a CGI program to process the valu
form.
The sportsHeader dynamic tag creates a star
for the HTML page as well as standard openit
<?sportsHeader title="Customer Query">
1305
As its action. the form must specify the Web
Internet

Fig. 15C

querydb.html File

1525

This HTML page contains a form that invokes an app page instead of a CGI program to process the values in the form.

The sportsHeader dynamic tag creates a standard header for the HTML page as well as standard opening text.

1535

```
<?sportsHeader title="Customer Query">
```

1540

1530

As its action, the form must specify the Web Driver utility.

```
<P>
<FORM ACTION="<?MIVAR>$WEB_HOME<?/MIVAR" METHOD="GET">
```

1550

1530

To specify the app page, the form must use a hidden input component. The input component must have a name of **MIval** and a value that's the name of the app page. The input component below specifies the cust_db.html app page.

1535

```
<INPUT TYPE="HIDDEN" NAME="MIval" VALUE="/examples/CustRpt/cust_db.html">

Optional state:
<INPUT TYPE="TEXT" NAME="selectState" SIZE="3" MAXLENGTH="2">
<INPUT TYPE="SUBMIT" NAME="Submit" VALUE="Submit">

</FORM>
</P>


</BODY>
</HTML>
```

1530

Fig. 15D

```
<!-- <ibyx>
<intro>
<p><abstract>This HTML page contains a form that invokes
an app page</abstract> instead of a CGI program to process
the values in the form.
</p>
</intro>
</ibyx> -->

<!-- <ibyx>
<p>The sportsHeader dynamic tag creates a standard header
for the HTML page as well as standard opening text.
</p>
</ibyx> -->
<?sportsHeader title="Customer Query">

<!-- <ibyx>
<p>As its action, the form must specify the Web Driver utility.
</p>
</ibyx> -->
<P>
<FORM ACTION="<?MIVAR>$WEB_HOME<?/MIVAR>" METHOD="GET">

<!-- <ibyx>
<p>To specify the app page, the form must use a hidden input component.
The input component must have a name of <strong>MIval</strong> and
a value that's the name of the app page.  The input component below
specifies the <a href="cust_db.html">cust_db.html</a> app page.
</p>
</ibyx> -->
<INPUT TYPE="HIDDEN" NAME="MIval" VALUE="/examples/CustRpt/cust_db.html">

Optional state:
<INPUT TYPE="TEXT" NAME="selectState" SIZE="3" MAXLENGTH="2">
<INPUT TYPE="SUBMIT" NAME="Submit" VALUE="Submit">

</FORM>
</P>

<?annotate>

</BODY>
</HTML>
```

1535

1560

1535

Fig. 15E

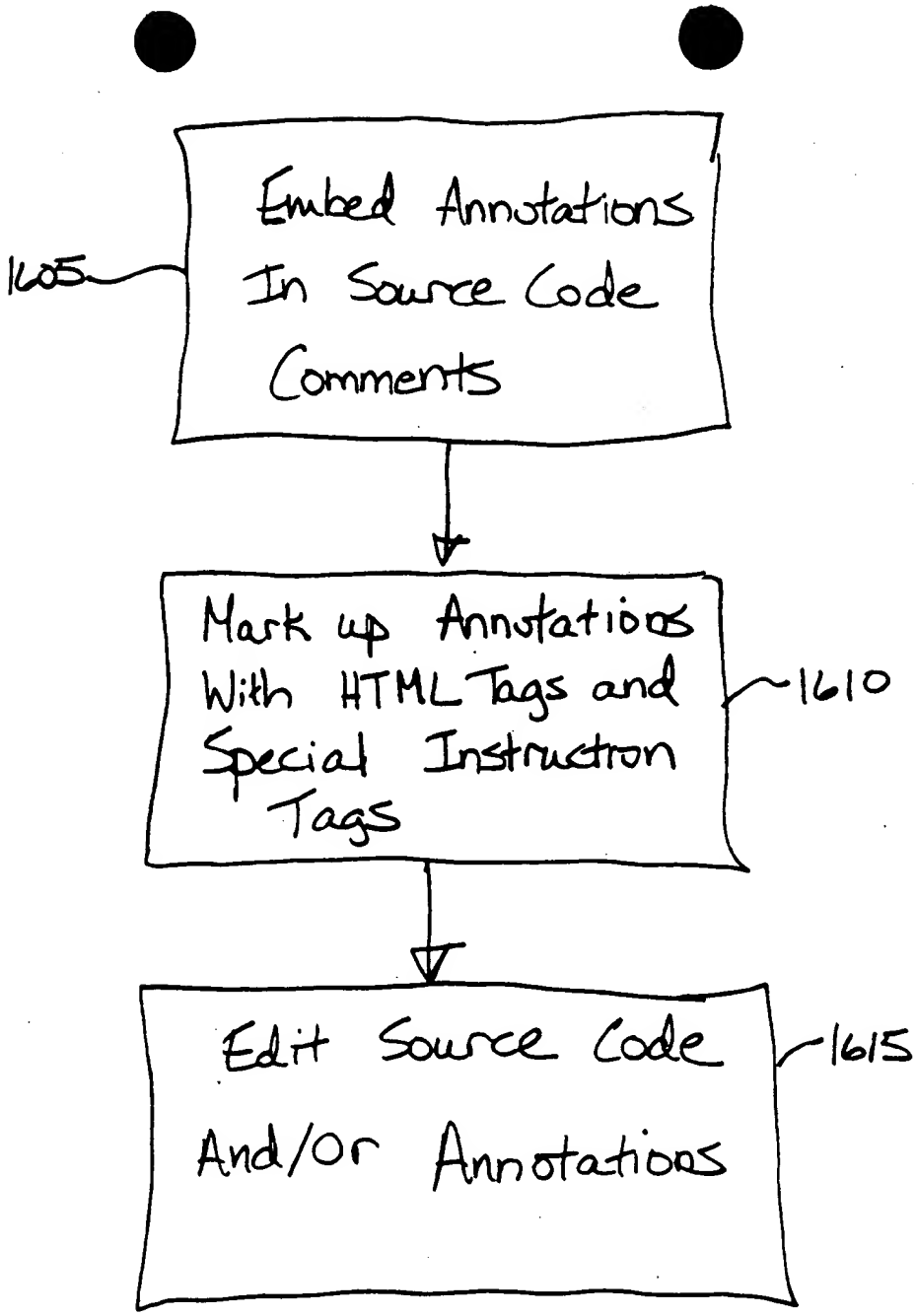

Embed Annotations In Source Code Comments
1605
Mark up Annotations With HTML Tags and Special Instruction Tags
1610
Edit Source Code And/Or Annotations
1615

Fig. 16A

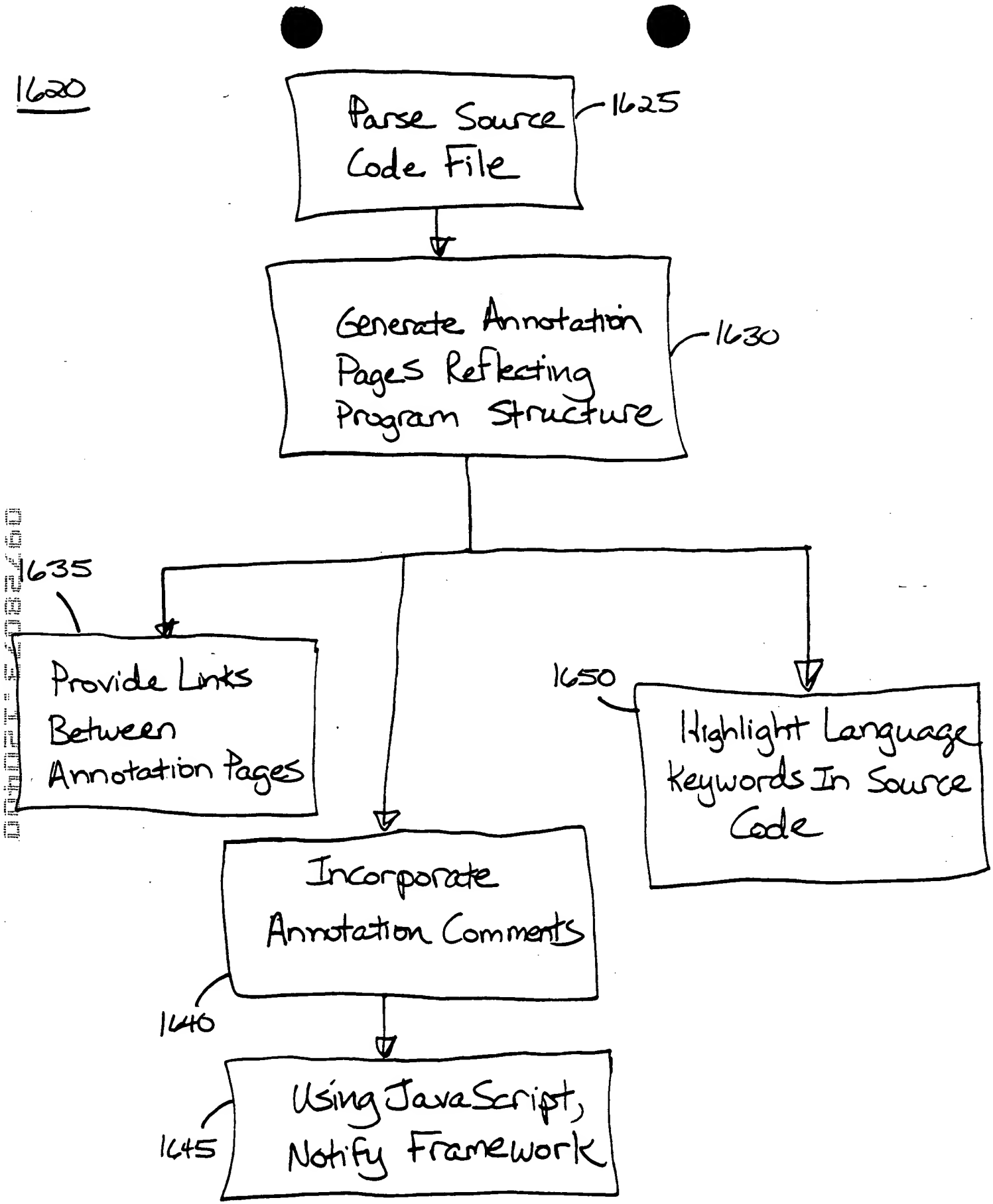
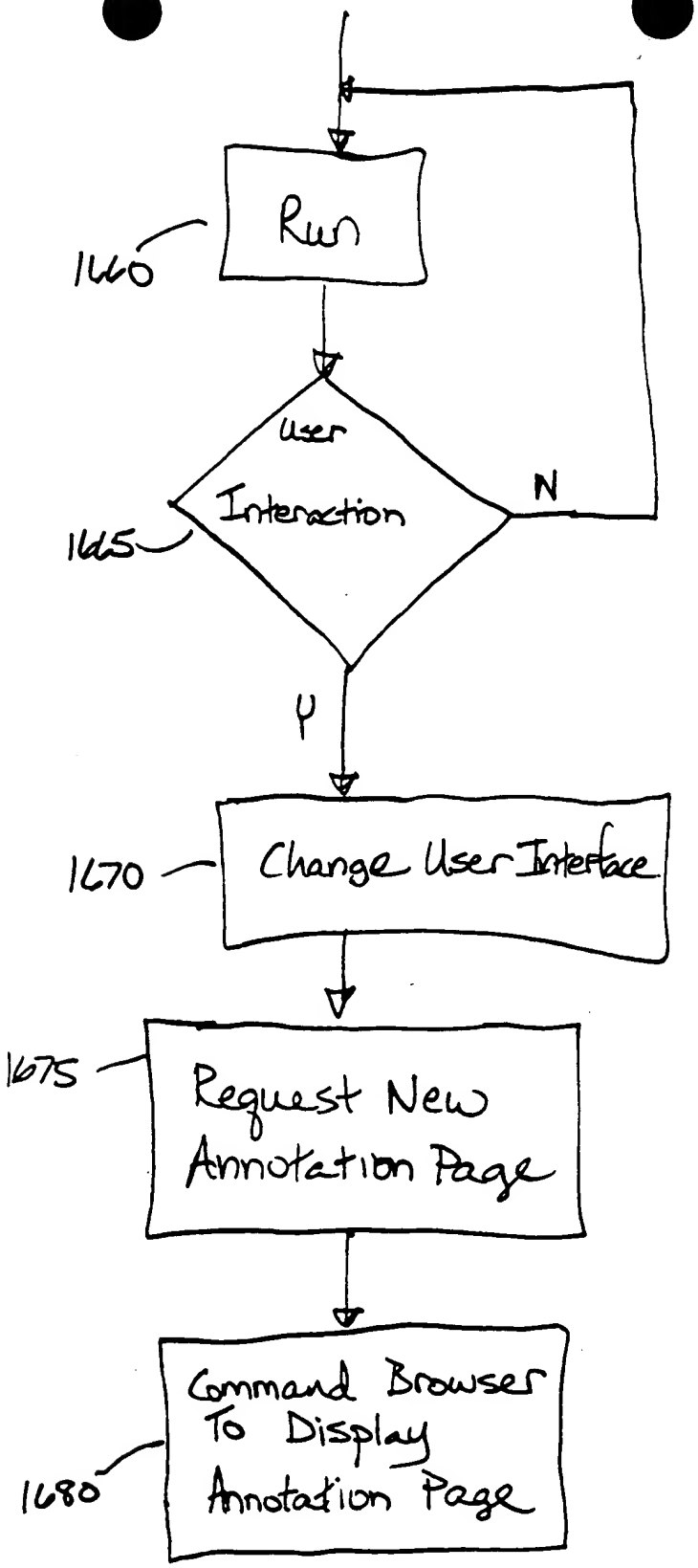

1620
Parse Source Code File
1625
Generate Annotation Pages Reflecting Program Structure
1630
1635
Provide Links Between Annotation Pages
1650
Highlight Language Keywords In Source Code
Incorporate Annotation Comments
1640
Using JavaScript, Notify Framework
1645

Fig. 16B


Run
1660
User
Interaction
1665
N
Y
1670
Change User Interface
1675
Request New
Annotation Page
Command Browser
To Display
Annotation Page
1680

1655

Fig. 16C

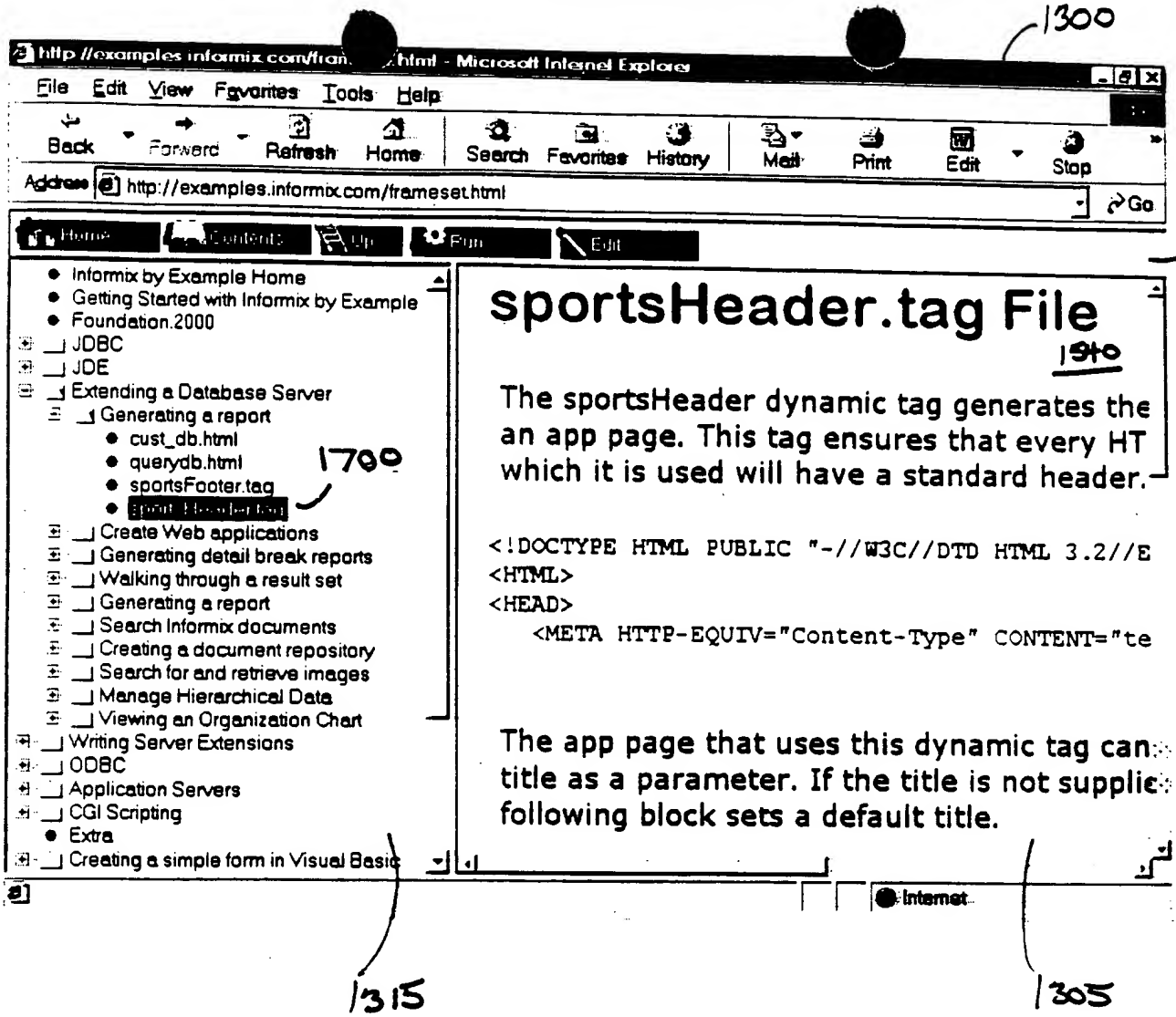

1300
http://examples.informix.com/frameset.html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Refresh Home Search Favorites History Mail Print Edit Stop
Address http://examples.informix.com/frameset.html
Go
Home Contents Up Print Edit
1310
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
JDBC
JDE
Extending a Database Server
Generating a report
cust_db.html
querydb.html
sportsFooter.tag
sportsHeader.tag
1700
Create Web applications
Generating detail break reports
Walking through a result set
Generating a report
Search Informix documents
Creating a document repository
Search for and retrieve images
Manage Hierarchical Data
Viewing an Organization Chart
Writing Server Extensions
ODBC
Application Servers
CGI Scripting
Extra
Creating a simple form in Visual Basic
sportsHeader.tag File
1910
The sportsHeader dynamic tag generates the
an app page. This tag ensures that every HT
which it is used will have a standard header.
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//E
<HTML>
<HEAD>
<META HTTP-EQUIV="Content-Type" CONTENT="te
The app page that uses this dynamic tag can
title as a parameter. If the title is not supplie
following block sets a default title.
Internet
1315
1305

Fig. 17A

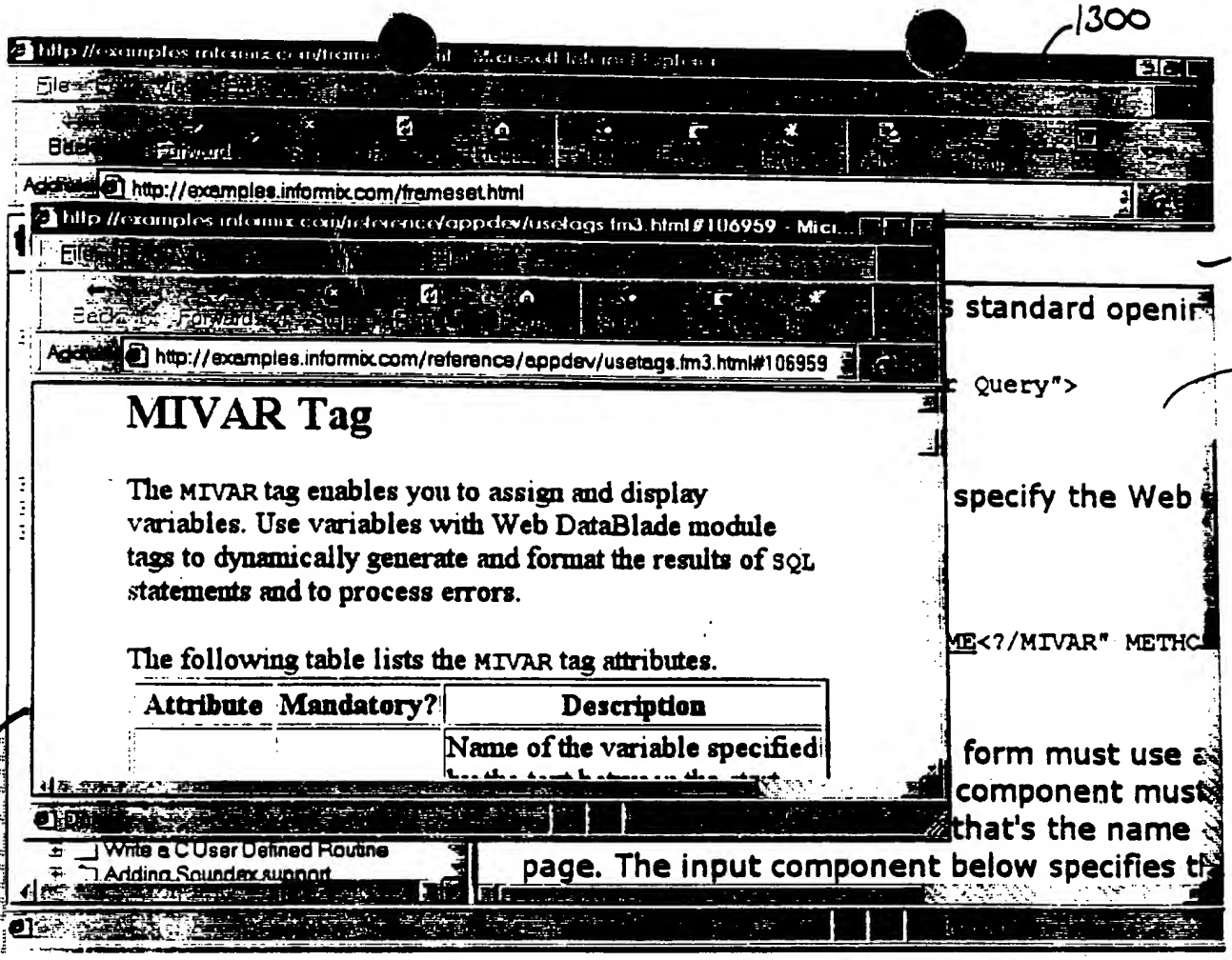

1300
http://examples.informix.com/frameset.html
http://examples.informix.com/reference/appdev/usetags.fm3.html#106959
1310
1305
1705
MIVAR Tag
The MIVAR tag enables you to assign and display variables. Use variables with Web DataBlade module tags to dynamically generate and format the results of SQL statements and to process errors.
The following table lists the MIVAR tag attributes.
Attribute
Mandatory?
Description
Name of the variable specified
standard openir
Query">
specify the Web
ME<?/MIVAR" METHC
form must use a
component must
that's the name
page. The input component below specifies th
Write a C User Defined Routine
Adding Soundex support

Fig. 17B

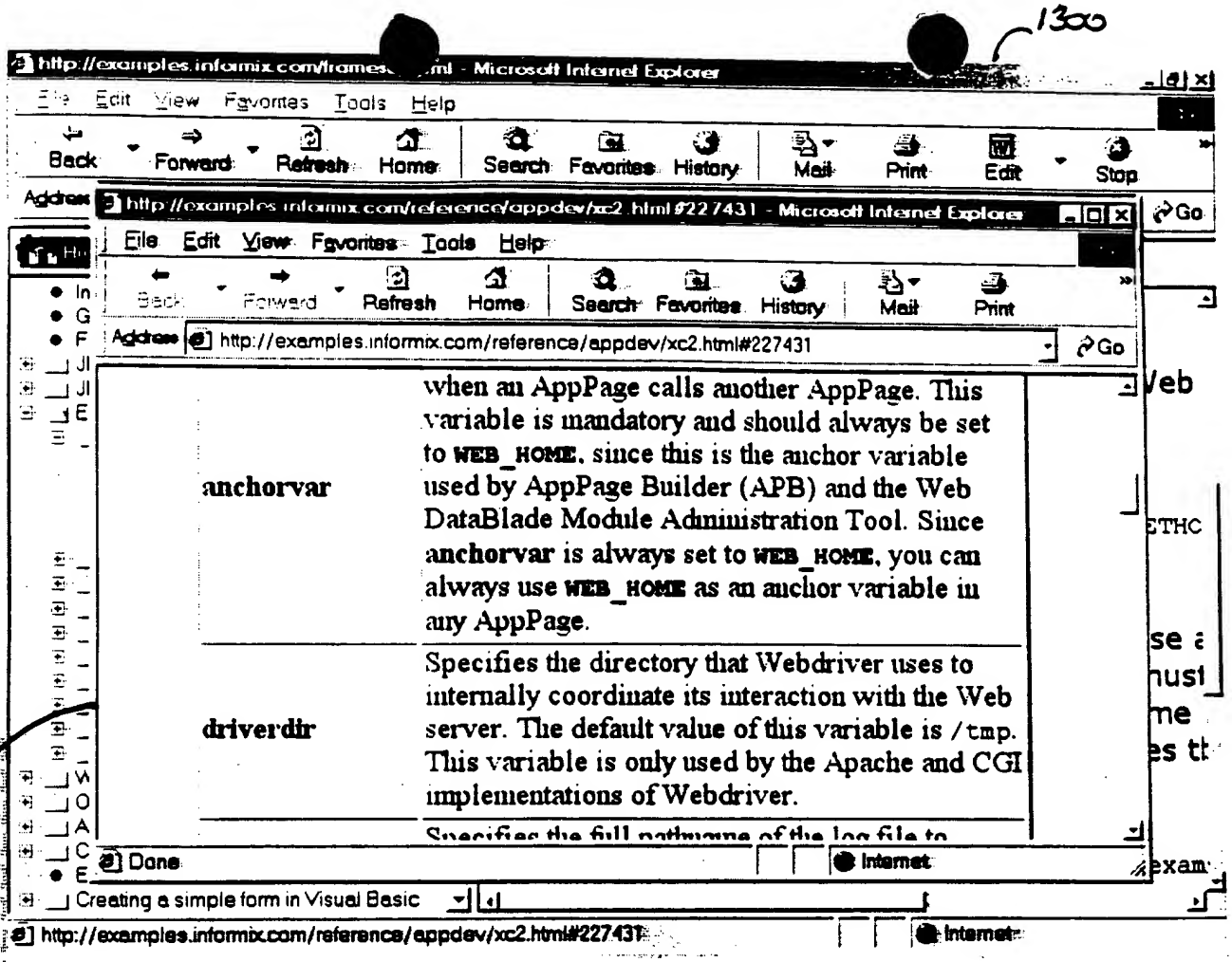

1300
http://examples.informix.com/frames...ml - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Refresh Home Search Favorites History Mail Print Edit Stop
Address
http://examples.informix.com/reference/appdev/xc2.html#227431 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Refresh Home Search Favorites History Mail Print
Address http://examples.informix.com/reference/appdev/xc2.html#227431
Go
when an AppPage calls another AppPage. This variable is mandatory and should always be set to WEB_HOME, since this is the anchor variable used by AppPage Builder (APB) and the Web DataBlade Module Administration Tool. Since anchorvar is always set to WEB_HOME, you can always use WEB_HOME as an anchor variable in any AppPage.
anchorvar
Specifies the directory that Webdriver uses to internally coordinate its interaction with the Web server. The default value of this variable is /tmp. This variable is only used by the Apache and CGI implementations of Webdriver.
driverdir
Done
Internet
Creating a simple form in Visual Basic
http://examples.informix.com/reference/appdev/xc2.html#227431
Internet
1710

Fig. 17C

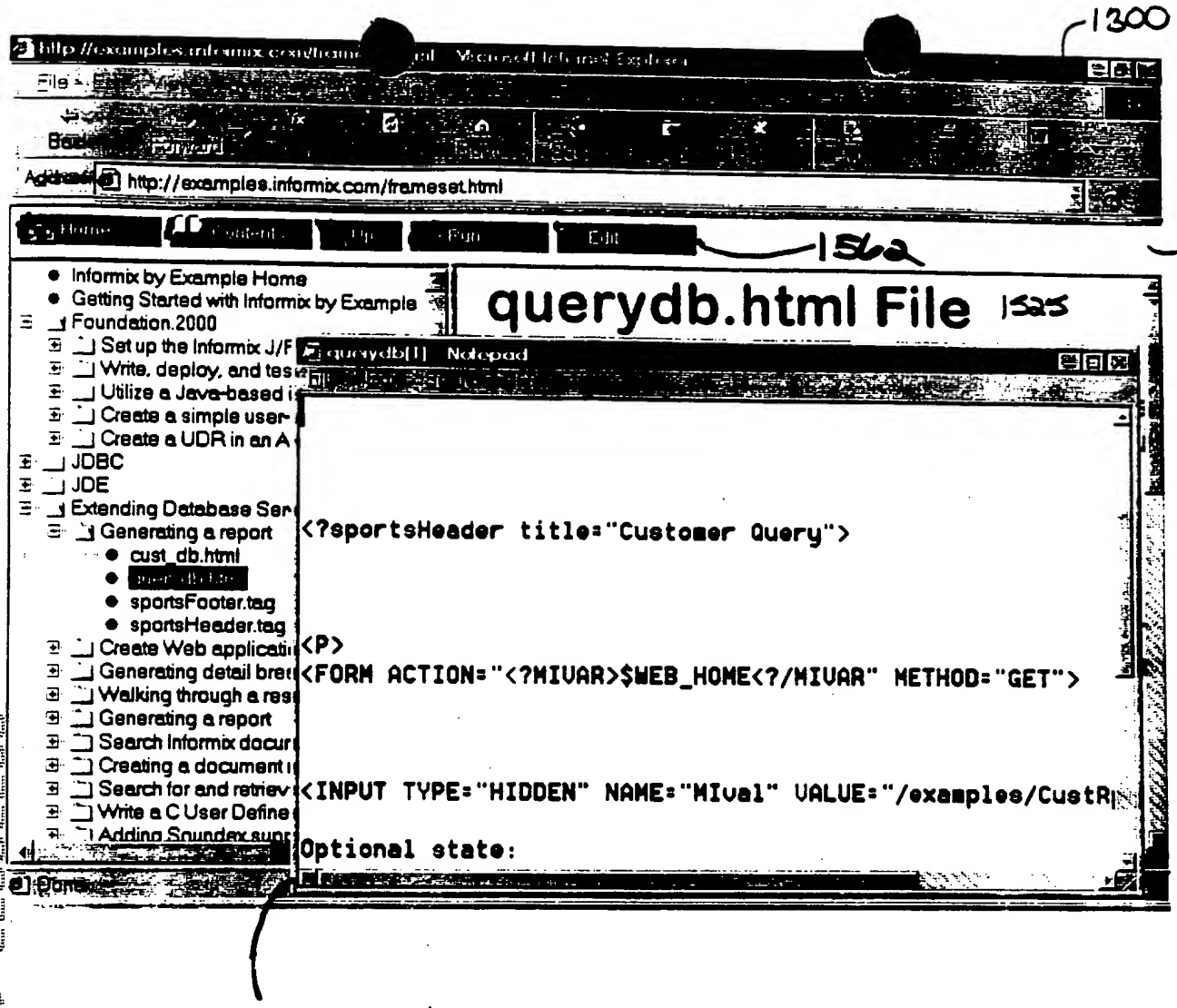

1300
http://examples.informix.com/frameset.html
Home
Edit
1562
1310
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
Set up the Informix J/F
Write, deploy, and tes
Utilize a Java-based i
Create a simple user-
Create a UDR in an A
JDBC
JDE
Extending Database Ser
Generating a report
cust_db.html
sportsFooter.tag
sportsHeader.tag
Create Web applicati
Generating detail bre
Walking through a res
Generating a report
Search Informix docur
Creating a document
Search for and retriev
Write a C User Define
querydb.html File
1525
querydb[1] Notepad
<?sportsHeader title="Customer Query">
<P>
<FORM ACTION="<?MIVAR>$WEB_HOME<?/MIVAR" METHOD="GET">
<INPUT TYPE="HIDDEN" NAME="MIval" VALUE="/examples/CustR
Optional state:
1715

Fig. 17D

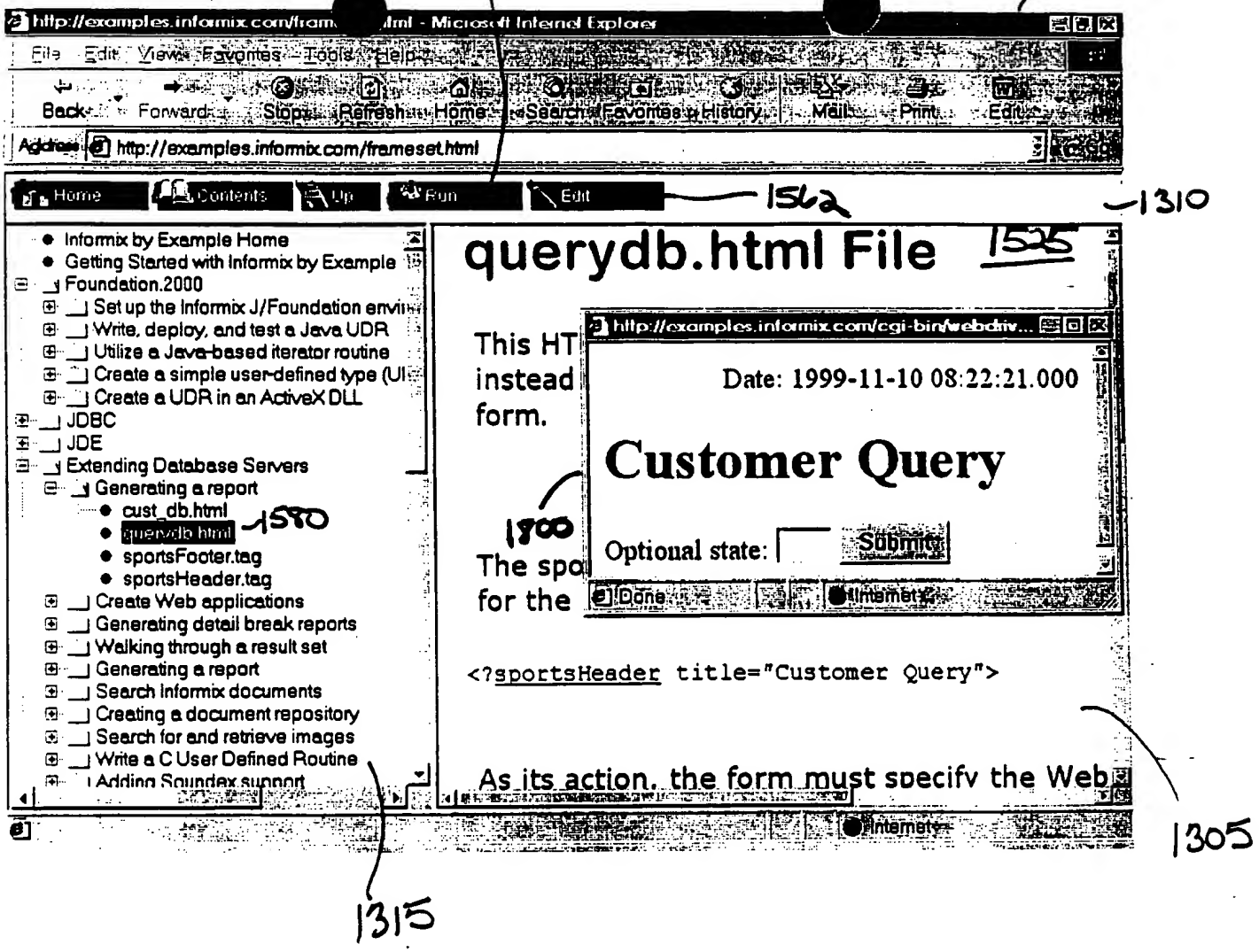

1300
http://examples.informix.com/fram...html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit
Address http://examples.informix.com/frameset.html
Home Contents Up Run Edit
1562
1310
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
Set up the Informix J/Foundation envi
Write, deploy, and test a Java UDR
Utilize a Java-based iterator routine
Create a simple user-defined type (UI
Create a UDR in an ActiveX DLL
JDBC
JDE
Extending Database Servers
Generating a report
cust_db.html
querydb.html
1580
sportsFooter.tag
sportsHeader.tag
Create Web applications
Generating detail break reports
Walking through a result set
Generating a report
Search Informix documents
Creating a document repository
Search for and retrieve images
Write a C User Defined Routine
Adding Soundex support
querydb.html File
1525
This HT
instead
form.
1800
The spo
for the
http://examples.informix.com/cgi-bin/webdriv...
Date: 1999-11-10 08:22:21.000
Customer Query
Optional state:
Submit
Done
Internet
<?sportsHeader title="Customer Query">
As its action, the form must specify the Web
Internet
1305
1315

Fig. 18A

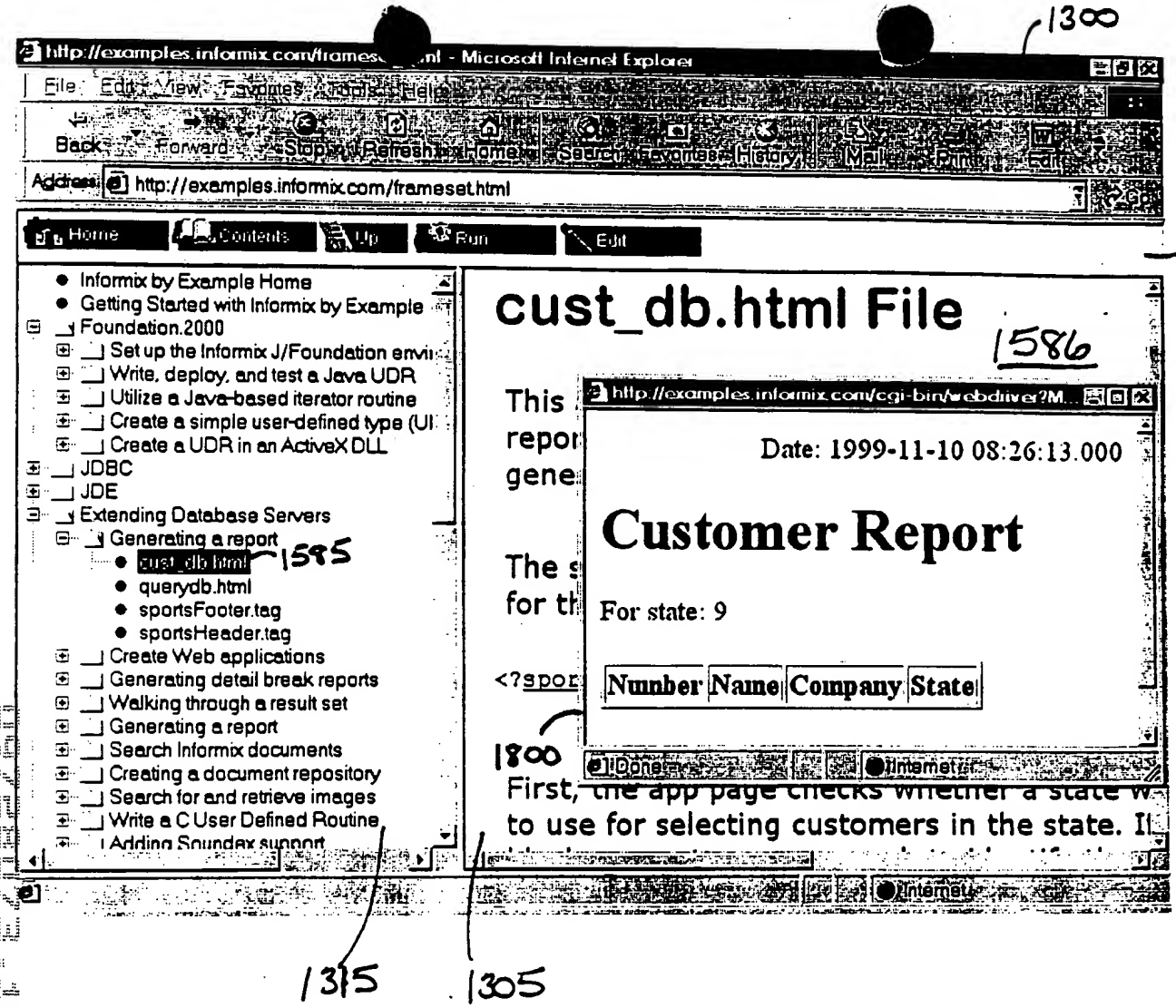

1300
http://examples.informix.com/frameset.html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit
Address http://examples.informix.com/frameset.html
Home Contents Up Run Edit
1310
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
Set up the Informix J/Foundation envi
Write, deploy, and test a Java UDR
Utilize a Java-based iterator routine
Create a simple user-defined type (UI
Create a UDR in an ActiveX DLL
JDBC
JDE
Extending Database Servers
Generating a report
cust_db.html
1595
querydb.html
sportsFooter.tag
sportsHeader.tag
Create Web applications
Generating detail break reports
Walking through a result set
Generating a report
Search Informix documents
Creating a document repository
Search for and retrieve images
Write a C User Defined Routine
Adding Soundex support
cust_db.html File
1586
This
repor
gene
The s
for th
<?spor
1800
First, the app page checks whether a state w
to use for selecting customers in the state. It
http://examples.informix.com/cgi-bin/webdriver?M...
Date: 1999-11-10 08:26:13.000
Customer Report
For state: 9
Number Name Company State
Done
Internet
1315
1305

Fig. 18B

```
<!-- <ibyx>
<intro>
<p><abstract>This app page accepts a query and generates an HTML
report</abstract> in response.
The app page uses dynamic tags to generate the header and footer for the
HTML report.
</p>
</intro>
</ibyx> -->
<!-- <ibyx>
<p>The sportsHeader dynamic tag creates a standard header
for the HTML page as well as standard opening text.
</p>
</ibyx> -->
<?sportsHeader title="Customer Report">

<!-- <ibyx>
<p>First, the app page checks whether a state was specified to use for
selecting customers in the state.  If so, the block generates a
paragraph to identify the state.
</p>
</ibyx> -->
<?MIVAR NAME=$WHERE_STR><?/MIVAR>
<?MIBLOCK COND="$(AND,$(XST,$selectState),$(<,0,$(STRLEN,$selectState)))">
        <?MIVAR NAME=$WHERE_STR>WHERE state="$selectState"<?/MIVAR>
        <?MIVAR><P>For state: $selectState</P><?/MIVAR>
<?/MIBLOCK>

<!-- <ibyx>
<p>Next, the app page starts the table that will contain the data.
</p>
</ibyx> -->
        <P><TABLE BORDER="1">
                <TR>
                <TH>Number</TH><TH>Name</TH><TH>Company</TH><TH>State</TH>
                </TR>

<!-- <ibyx>
<p>The MISQL block queries for customers, optionally selecting only customers
from the specified state.  Because the contents of the block are generated
for every row of data, a new table row describes each customer.

The &amp;amp;nbsp; HTML entity is a non-breaking space.  By putting a non-breaking
space in each column, we force the Web Browser to display the column even
if the value is null.
</p>
</ibyx> -->.
<?MISQL SQL="SELECT customer_num, fname, lname, company, state FROM customer $WHERE_STR;">
                <TR>
                <TD>$1 </TD><TD>$2 $3</TD><TD>$4 </TD><TD>$5 </TD>
                </TR>
<?/MISQL>

        </TABLE></P>

<!-- <ibyx>
<p>The sportsFooter dynamic tag creates a standard footer
for the HTML page.
</p>
</ibyx> -->
<?sportsFooter>
```

1810

Fig. 18C

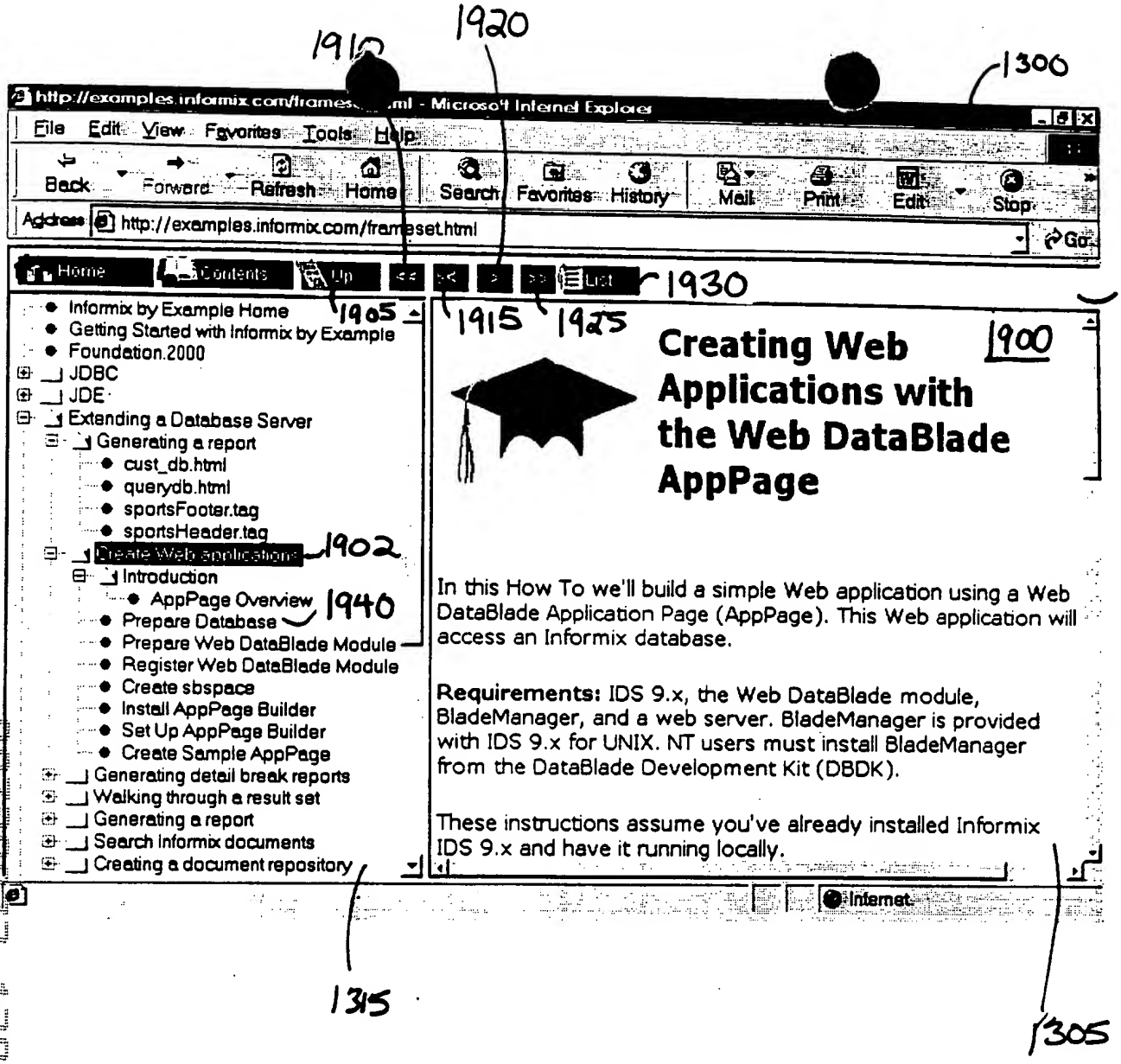

1910
1920
1300
http://examples.informix.com/frames...ml - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Refresh Home Search Favorites History Mail Print Edit Stop
Address http://examples.informix.com/frameset.html
Go
Home Contents Up
1930
1310
1905
1915
1925
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
JDBC
JDE
Extending a Database Server
Generating a report
cust_db.html
querydb.html
sportsFooter.tag
sportsHeader.tag
Create Web applications
1902
Introduction
AppPage Overview
1940
Prepare Database
Prepare Web DataBlade Module
Register Web DataBlade Module
Create sbspace
Install AppPage Builder
Set Up AppPage Builder
Create Sample AppPage
Generating detail break reports
Walking through a result set
Generating a report
Search Informix documents
Creating a document repository
1900
Creating Web Applications with the Web DataBlade AppPage
In this How To we'll build a simple Web application using a Web DataBlade Application Page (AppPage). This Web application will access an Informix database.
Requirements: IDS 9.x, the Web DataBlade module, BladeManager, and a web server. BladeManager is provided with IDS 9.x for UNIX. NT users must install BladeManager from the DataBlade Development Kit (DBDK).
These instructions assume you've already installed Informix IDS 9.x and have it running locally.
Internet
1315
1305

Fig. 19A

Creating Web Applications with the Web DataBlade AppPage

1930 1900

In this How To we'll build a simple Web application using a Web DataBlade Application Page (AppPage). This Web application will access an Informix database.

Requirements: IDS 9.x, the Web DataBlade module, BladeManager, and a web server. BladeManager is provided with IDS 9.x for UNIX. NT users must install BladeManager from the DataBlade Development Kit (DBDK).

These instructions assume you've already installed Informix IDS 9.x and have it running locally.

1935

Define a server connection and prepare a sample database.

1. Define a server connection with setnet32 (NT). Create a sample database or use the stores7 demo database. ▶ Prepare Database.

Prepare the Web DataBlade development environment.

2. Install the Web DataBlade module and BladeManager. ▶ Prepare Web DataBlade Development Environment.

3. Register the Web DataBlade module in the demo database with BladeManager. ▶ Register the Web DataBlade.

4. Create a sbspace for smart large objects, like gifs. ▶ Create Smart Blob Space (sbspace).

5. Install AppPage Builder in your database. ▶ Install AppPage Builder in Your Database.

6. Setup AppPage Builder on your web server. ▶ Setup AppPage Builder on Your Web Server.

7. Create a sample AppPage. ▶ Create Sample AppPage.

Run the sample application.

8. Enter the URL `http://your_server/scripts/webdriver.exe`.

This How To has been compiled into two separate files for ease of printing. The basic file contains all of the steps you need to Create Web Applications with AppPage Builder. The secondary file contains additional detailed instructions for setting and testing database environment properties.

Fig. 19B

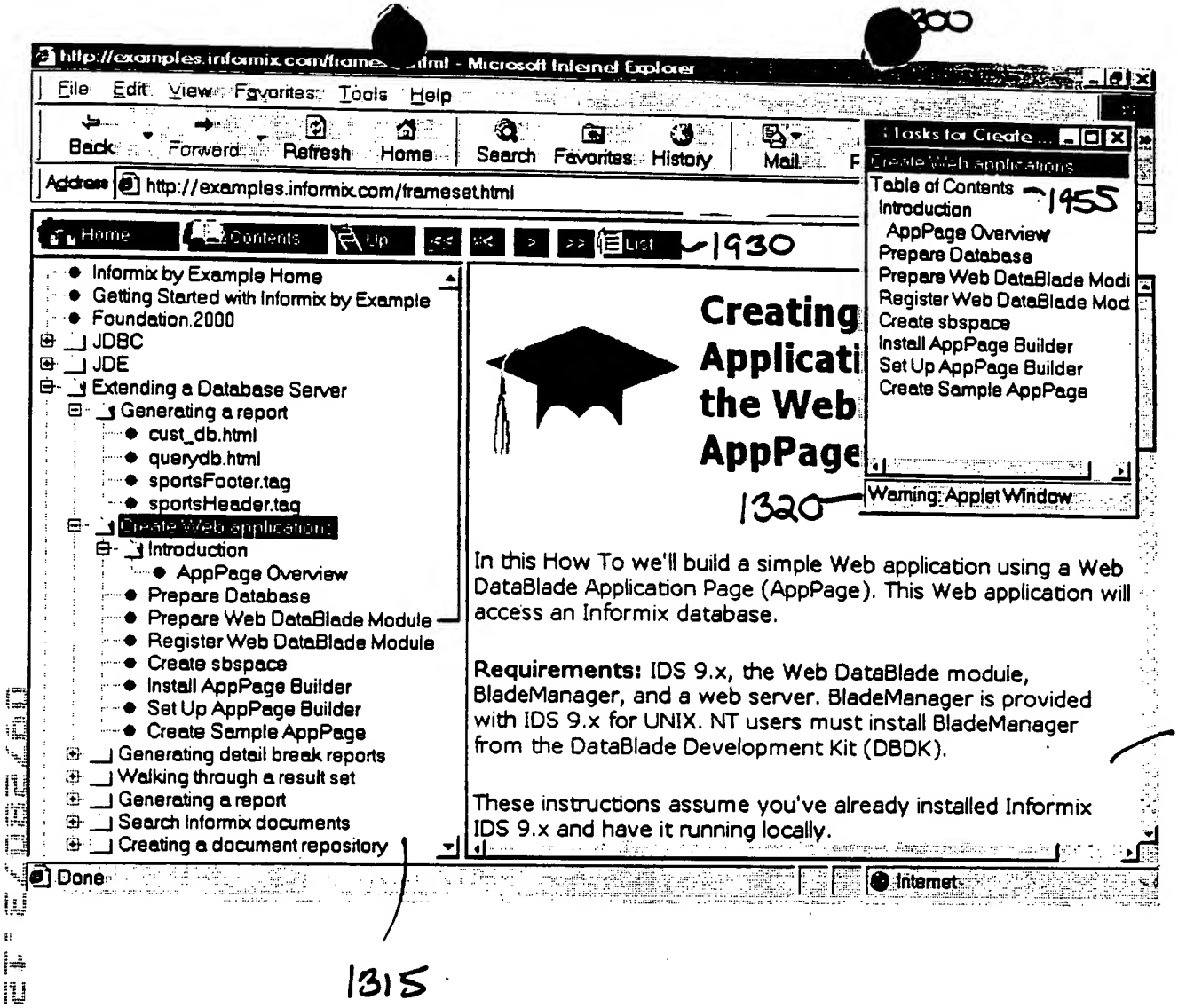

300
http://examples.informix.com/frame...html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Refresh Home Search Favorites History Mail
Address http://examples.informix.com/frameset.html
Tasks for Create...
Create Web applications
Table of Contents
Introduction
AppPage Overview
Prepare Database
Prepare Web DataBlade Modi
Register Web DataBlade Mod
Create sbspace
Install AppPage Builder
Set Up AppPage Builder
Create Sample AppPage
1955
Home Contents Up List
1930
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
JDBC
JDE
Extending a Database Server
Generating a report
cust_db.html
querydb.html
sportsFooter.tag
sportsHeader.tag
Create Web applications
Introduction
AppPage Overview
Prepare Database
Prepare Web DataBlade Module
Register Web DataBlade Module
Create sbspace
Install AppPage Builder
Set Up AppPage Builder
Create Sample AppPage
Generating detail break reports
Walking through a result set
Generating a report
Search Informix documents
Creating a document repository
Creating
Applicati
the Web
AppPage
1320
Warning: Applet Window
In this How To we'll build a simple Web application using a Web DataBlade Application Page (AppPage). This Web application will access an Informix database.
Requirements: IDS 9.x, the Web DataBlade module, BladeManager, and a web server. BladeManager is provided with IDS 9.x for UNIX. NT users must install BladeManager from the DataBlade Development Kit (DBDK).
1305
These instructions assume you've already installed Informix IDS 9.x and have it running locally.
Done
Internet
1315

Fig. 19C

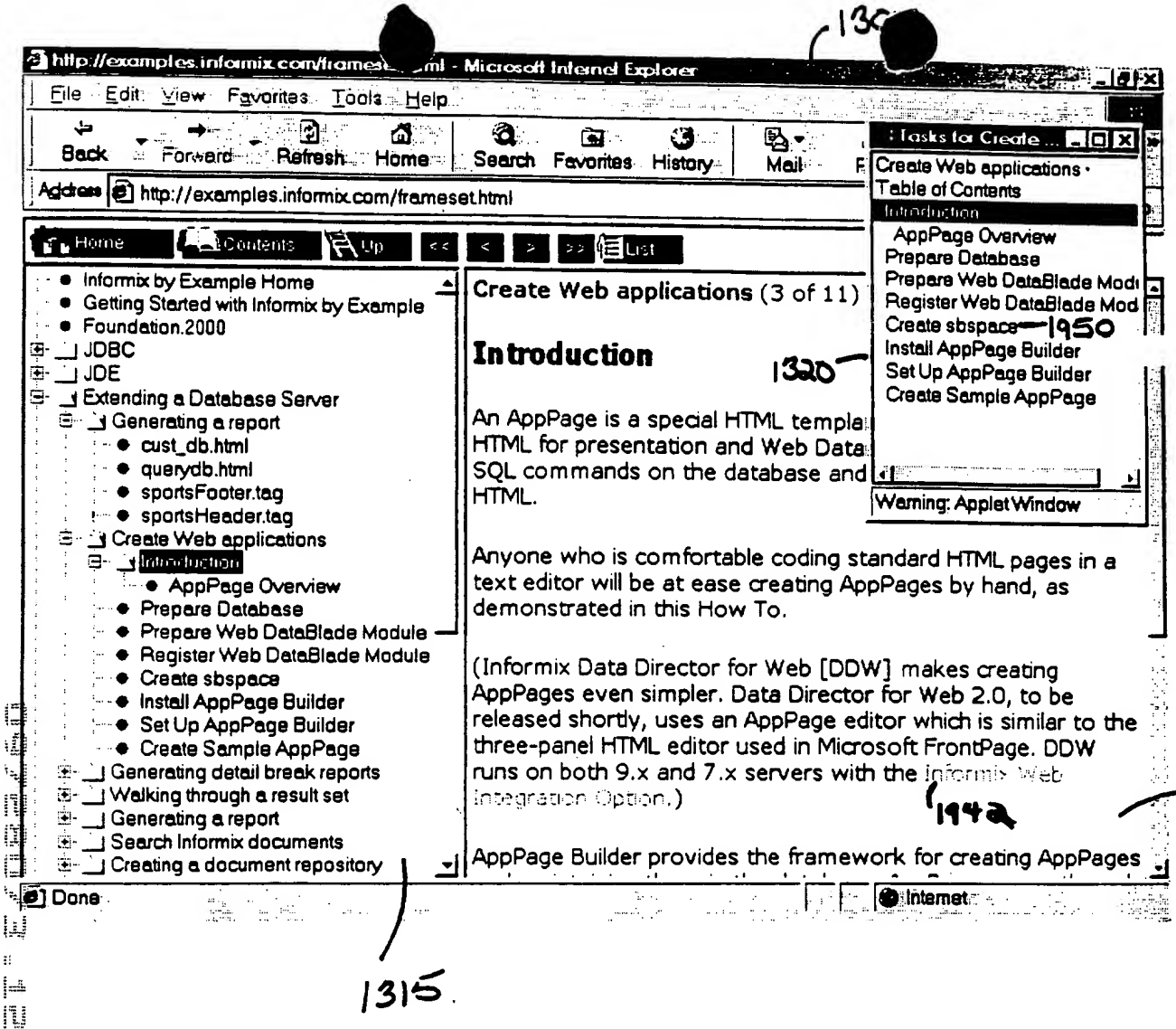

1350
http://examples.informix.com/frameset.html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Refresh Home Search Favorites History Mail
Address http://examples.informix.com/frameset.html
Tasks for Create ...
Create Web applications
Table of Contents
Introduction
AppPage Overview
Prepare Database
Prepare Web DataBlade Mod
Register Web DataBlade Mod
Create sbspace
Install AppPage Builder
Set Up AppPage Builder
Create Sample AppPage
1950
Warning: Applet Window
Home Contents Up << < > >> List
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
JDBC
JDE
Extending a Database Server
Generating a report
cust_db.html
querydb.html
sportsFooter.tag
sportsHeader.tag
Create Web applications
Introduction
AppPage Overview
Prepare Database
Prepare Web DataBlade Module
Register Web DataBlade Module
Create sbspace
Install AppPage Builder
Set Up AppPage Builder
Create Sample AppPage
Generating detail break reports
Walking through a result set
Generating a report
Search Informix documents
Creating a document repository
1315
Create Web applications (3 of 11)
Introduction
1320
An AppPage is a special HTML templa
HTML for presentation and Web Data
SQL commands on the database and
HTML.
Anyone who is comfortable coding standard HTML pages in a text editor will be at ease creating AppPages by hand, as demonstrated in this How To.
(Informix Data Director for Web [DDW] makes creating AppPages even simpler. Data Director for Web 2.0, to be released shortly, uses an AppPage editor which is similar to the three-panel HTML editor used in Microsoft FrontPage. DDW runs on both 9.x and 7.x servers with the Informix Web Integration Option.)
1942
1305
AppPage Builder provides the framework for creating AppPages
Done
Internet

Fig. 19D

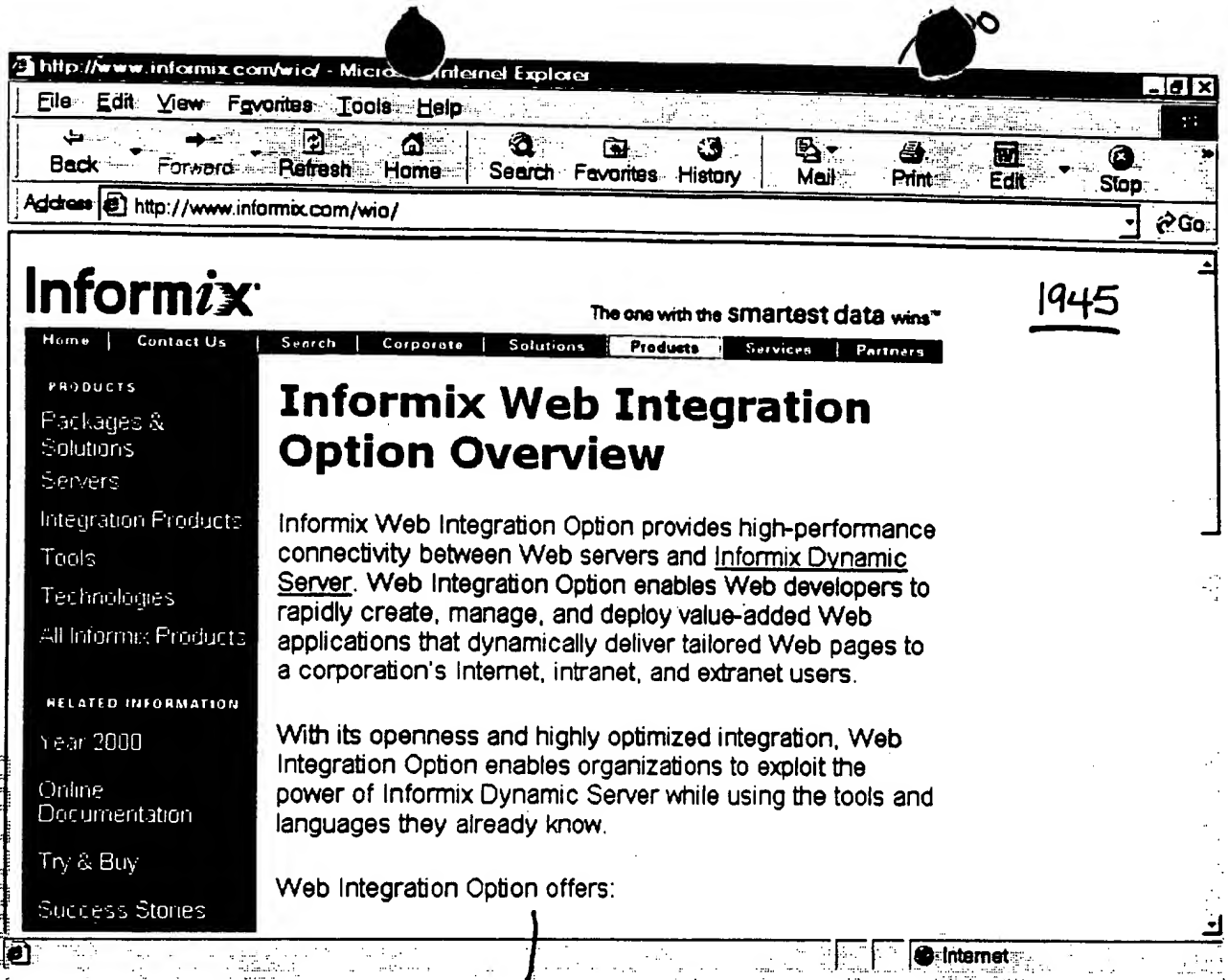

http://www.informix.com/wio/ - Micro Internet Explorer
File Edit View Favorites Tools Help
Back Forward Refresh Home Search Favorites History Mail Print Edit Stop
Address http://www.informix.com/wio/
Go
Informix
The one with the smartest data wins™
1945
Home | Contact Us | Search | Corporate | Solutions | Products | Services | Partners
PRODUCTS
Packages & Solutions
Servers
Integration Products
Tools
Technologies
All Informix Products
RELATED INFORMATION
Year 2000
Online Documentation
Try & Buy
Success Stories
Informix Web Integration Option Overview
Informix Web Integration Option provides high-performance connectivity between Web servers and Informix Dynamic Server. Web Integration Option enables Web developers to rapidly create, manage, and deploy value-added Web applications that dynamically deliver tailored Web pages to a corporation's Internet, intranet, and extranet users.
With its openness and highly optimized integration, Web Integration Option enables organizations to exploit the power of Informix Dynamic Server while using the tools and languages they already know.
Web Integration Option offers:
Internet
1305

Fig. 19E

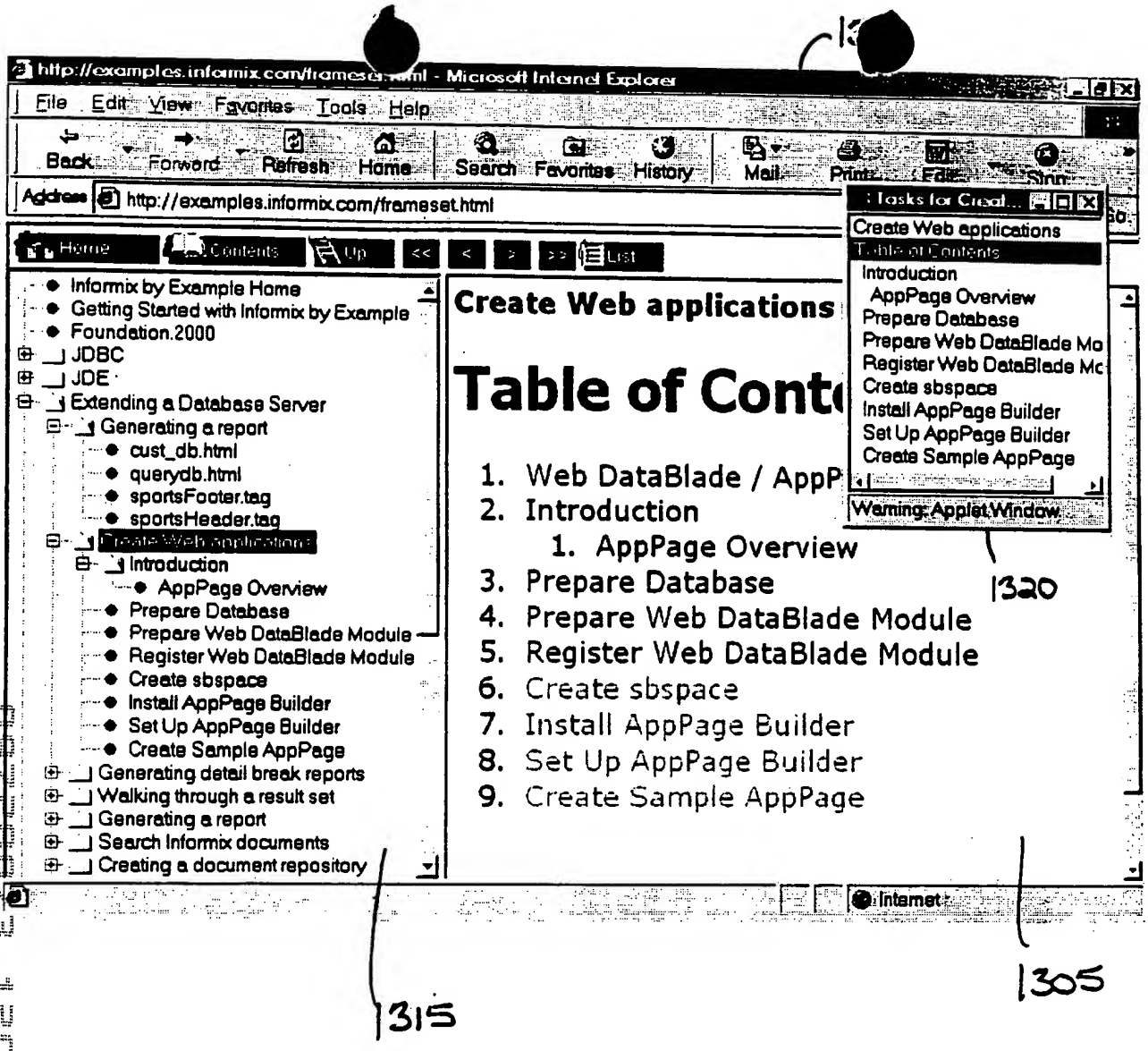

http://examples.informix.com/frameset.html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Refresh Home Search Favorites History Mail Print Edit Discuss
Address http://examples.informix.com/frameset.html
Home Contents Up << < > >> List
Informix by Example Home
Getting Started with Informix by Example
Foundation.2000
JDBC
JDE
Extending a Database Server
Generating a report
cust_db.html
querydb.html
sportsFooter.tag
sportsHeader.tag
Create Web applications
Introduction
AppPage Overview
Prepare Database
Prepare Web DataBlade Module
Register Web DataBlade Module
Create sbspace
Install AppPage Builder
Set Up AppPage Builder
Create Sample AppPage
Generating detail break reports
Walking through a result set
Generating a report
Search Informix documents
Creating a document repository
Create Web applications
Table of Conte
1. Web DataBlade / AppP
2. Introduction
1. AppPage Overview
3. Prepare Database
4. Prepare Web DataBlade Module
5. Register Web DataBlade Module
6. Create sbspace
7. Install AppPage Builder
8. Set Up AppPage Builder
9. Create Sample AppPage
Tasks for Creat...
Create Web applications
Table of Contents
Introduction
AppPage Overview
Prepare Database
Prepare Web DataBlade Mo
Register Web DataBlade Mc
Create sbspace
Install AppPage Builder
Set Up AppPage Builder
Create Sample AppPage
Warning: Applet Window
Internet
1320
1305
1315

Fig. 19F

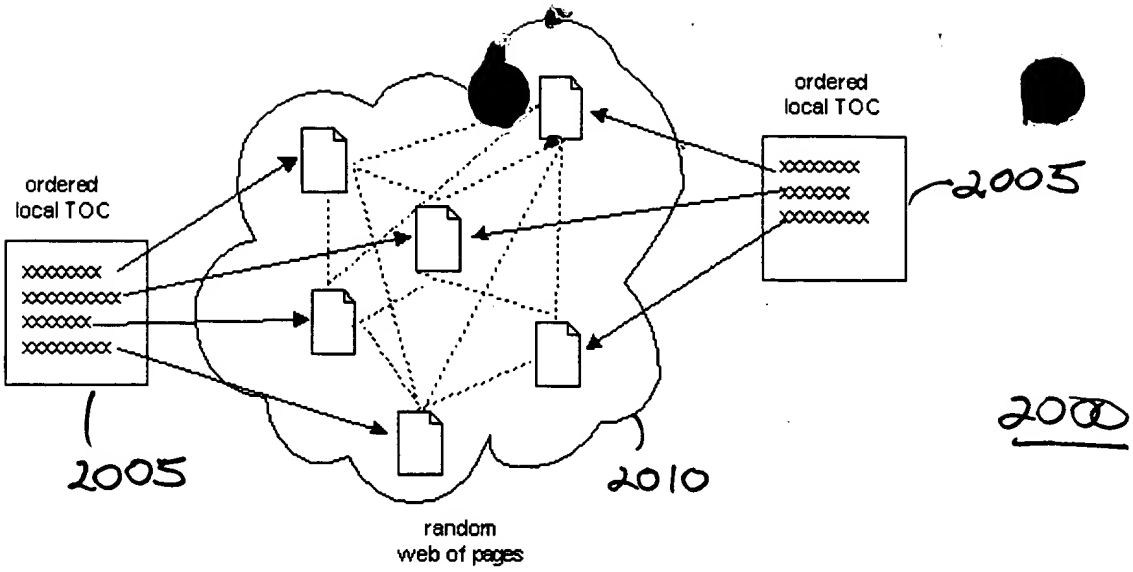


Fig. 20